ServletExec™ 4.1

Installation Guide

for Microsoft Internet Information Server
Netscape Enterprise Server
iPlanet Web Server
and Apache HTTP Server

ServletExec™ 4.1 Installation Guide

Contents

1

1. Introduction

Installing ServletExec 4.1 (all products)

New Atlanta ServletExec 4.1 is a high-performance, reliable, and cost-effective web application server that implements the Java™ Servlet API 2.3 and JavaServer Pages™ (JSP) 1.2 standards defined by Sun Microsystems, Inc. as component technologies of the Java 2 Platform, Enterprise Edition (J2EE™). Additional information about J2EE, servlets, and JSP technologies can be found on Sun's web site:

http://java.sun.com/j2ee/

ServletExec enables you to deploy servlets and JSP on Microsoft® Internet Information Server (IIS), Netscape® Enterprise Server (NES), iPlanet™ Web Server (iWS), and Apache HTTP Server in a standard, robust, high-performance environment.

1.1 ServletExec Products

ServletExec supports two basic configurations (in-process and out-of-process) with three products:

- **ServletExec 4.1 Servlet Engine**, an in-process web server plugin that adds high-performance servlet and JSP support to Microsoft IIS, Netscape Enterprise Server (NES), and iPlanet Web Server (iWS).
  - **ServletExec/ISAPI** is the plugin servlet engine for Microsoft IIS on Windows NT® or Windows® 2000.
  - **ServletExec/NSAPI** is the plugin servlet engine for NES/iWS on Windows NT/2000 and SPARC Solaris.
- **ServletExec 4.1 Application Server**, an out-of-process web application server that includes web server adapters for deployment with Microsoft IIS, NES/iWS, and Apache.
  - **ServletExec/AS** is available for Windows NT/2000 and various UNIX platforms, including SPARC Solaris, Linux, HP-UX, and AIX.

The key characteristics of each configuration are listed in Table 1. ServletExec Configurations.

If you're using Microsoft IIS or NES/iWS and you aren't sure whether the Servlet Engine or Application Server configuration is best for you, we recommend starting with the Servlet Engine version of ServletExec (ServletExec/ISAPI or ServletExec/NSAPI). If you're using Apache, only the ServletExec/AS version is currently available.

ServletExec Servlet Engine	ServletExec Application Server
• Runs within the web server process (loaded as a Windows DLL or UNIX `.so`)	• Runs as a standalone process, separate from the web server
• Best performance	• Greatest architectural flexibility
• Easiest to install and configure	• Enhanced safety (process isolation)
• Single Java VM per web server	• Multiple Java VMs per web server
• ServletExec starts/stops automatically when the web server starts/stops	• ServletExec starts/stops independently of the web server

Table 1. ServletExec Configurations

There are separate ServletExec installers based on your web server, operating system, and choice of Servlet Engine or Application Server product. Make sure you've downloaded the correct installer, and then refer to the appropriate chapter of this manual for detailed installation instructions.

1.2 Technical Support

If you're having difficulty installing or configuring ServletExec, check the online Tech Support FAQ:

> **http://www.newatlanta.com/support/support-faq.jsp**

You may also want to consider subscribing to the ServletExec-Interest mailing list, a user-supported discussion forum for ServletExec developers:

> **http://www.newatlanta.com/support/se-interest.jsp**

Details regarding free and paid support options, including online-, telephone-, and pager-based support are available from the ServletExec web site:

> **http://www.newatlanta.com/support.jsp**

2

2. ServletExec/ISAPI: Microsoft IIS Plugin (Windows)

Installing the plugin servlet engine for Microsoft Internet Information Server

ServletExec/ISAPI is a plugin servlet engine for Microsoft Internet Information Server (IIS) on Microsoft Windows NT® and Microsoft Windows 2000®. ServletExec/ISAPI enables you to deploy Java Servlets and JavaServer Pages (JSP) on Microsoft IIS in a standard, robust, high-performance environment.

The ServletExec/ISAPI installer is named `ServletExec_ISAPI_41.exe`. If you have not already done so, you can download this installer from:

http://www.newatlanta.com/downloads/index.jsp

This chapter contains important information that will enable you to verify your installation of ServletExec/ISAPI. It will also be useful if you ever need to uninstall ServletExec/ISAPI and ensure you've completely removed all installed components.

2.1 Upgrading from a Previous Version

You can upgrade to ServletExec 4.1 from all previous versions except versions 3.0C or 3.0E. If you have either of these versions installed, you will need to completely uninstall that version and do a clean installation of 4.1.

To upgrade to a new version of ServletExec/ISAPI and maintain your old configuration settings

1. Stop IIS.

   ▪ For IIS 4.0 on Windows NT 4.0: To completely stop IIS 4.0, you must either stop the IIS Admin Service from the **Services** Control Panel, or execute the `stop_iis.bat file`, which can be found in the `ServletExec ISAPI` directory.

   ▪ For IIS 5.0 on Windows 2000: To completely stop IIS 5.0, you can either: (a) stop the IIS Admin Service using the **Services/Administrative Tools** control panel, (b) execute the `stop_iis.bat` command file or, (c) click **Restart IIS** in the Internet Services Manager administrative tool.

2. Close all open applications, making sure to include DBMON.
3. Make backup copies of the `ServletExecData`, `Servlets`, `classes` (if you've added an additional one), and `webapps` (versions 3.1 and above only) subdirectories of the `ServletExec ISAPI` directory. The location of this directory varies depending upon your version of ServletExec.

    ▪ For ServletExec/ISAPI 2.2 and earlier: The default location for the `ServletExec ISAPI` directory is within the `C:\InetPub` directory.

    ▪ For ServletExec/ISAPI 3.0 and later: The default location for the `ServletExec ISAPI` directory is within the `C:\Program Files\New Atlanta` directory.

4. Uninstall the old version of ServletExec/ISAPI. Click the **Add/Remove** or **Change/Remove** control panel, depending on your operating system and follow the instructions on the **Install/Uninstall** tab.
5. Run the ServletExec/ISAPI 4.1 installer, installing into the same location as the previous installation.

    Note

    The default installation directory for versions 2.2 and earlier is different.

    Important

    For a successful upgrade, you must install everything into the same directory as the previous version.

6. Copy the `ServletExecData`, `Servlets`, `classes` (if you've added an additional one), and `webapps` (versions 3.1 and above only) subdirectories you made backup copies of in Step 2, to the new `ServletExec ISAPI` directory created by the installer in Step 5.
7. Restart IIS.

After restarting your web server, ServletExec/ISAPI 4.1 will run using your old configuration settings.

Tip

If you have any problems, you can restore the subdirectories from the backups you made in Step 3.

2.2 System Requirements

2.2.1 Operating System and Web Server

- ServletExec/ISAPI 4.1 is not supported on Windows 95, Windows 98 or Windows ME.
- ServletExec/ISAPI 4.1 supports the following operating systems and web servers:

OS	Web Server
Windows 2000	Internet Information Server (IIS) 5.0
Windows NT 4.0 Server	Internet Information Server (IIS) 4.0
Windows NT 4.0 Workstation	Peer Web Services (PWS) 4.0

Table 2. Supported Operating Systems and Web Servers

2.2.2 JDK or JRE

ServletExec/ISAPI 4.1 is not supported on JDK or JRE 1.1.x or earlier.

To install ServletExec/ISAPI 4.1, you must first install one of these Java Development Kit (JDK™) or Java™ Runtime Environment (JRE) choices:

- JDK/JRE 1.2.2 or 1.3.1 for Windows from Sun Microsystems
- JDK/JRE for Windows 1.3 from IBM

Important

The JRE does not contain a Java compiler, which is required by ServletExec to support JavaServer Pages (JSP). Be sure to install the JDK, and not the JRE, if you plan to use JSP.

You can download the Sun JDKs for Windows from:

http://java.sun.com/j2se/

You can download the IBM JDKs for Windows from:

http://www.ibm.com/developerworks/java/

As of this writing, the latest production versions of the Sun JDKs for Windows are 1.2.2 and 1.3.1. The latest production version of the IBM JDK for Windows is 1.3. These are the versions we tested and recommend using with ServletExec/ISAPI 4.1.

2.2.3 JDK/JRE Installation

To install ServletExec/ISAPI, you must have first installed a compatible version of the JDK or JRE as explained in Section 2.2.2.

Important

The JDK or JRE must be installed on a local drive and not on a mapped network drive. If the JDK/JRE has been installed on a mapped network drive, ServletExec/ISAPI will not be able to load and initialize the Java VM.

2.2.3.1 Multiple JDK/JRE Versions Installed

You may have multiple versions of the JDK/JRE installed on your system. ServletExec/ISAPI (including the installer) uses registry entries to determine which installed JDK/JRE to use. It will look for an installed JDK first and if it doesn't find one it will look for a JRE. It will look for JDK/JREs from vendors in the following order: Sun Microsystems, and then IBM. Here's the complete algorithm:

1. Look for a Sun JDK using the following registry key:
    `HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Development Kit`

    - If Sun JDK found, go to Step 2. Otherwise, look for a Sun JRE using the following key:

      `HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Runtime Environment`

    - If Sun JRE found, go to Step 2. Otherwise, look for an IBM JDK or JRE using the following keys .

      `HKEY_LOCAL_MACHINE\SOFTWARE\IBM\Java Development Kit`

      or…

      `HKEY_LOCAL_MACHINE\SOFTWARE\IBM\Java Runtime Environment`

    - If no Sun or IBM JDK/JRE found using the above registry keys, then abort ServletExec installer.

2. Read the `CurrentVersion` variable from the key found in Step 1. Currently, the only valid values for this variable start with "1.2" or "1.3" (including, for example "`1.2.2`").

3. Append the value of the `CurrentVersion` variable from Step 2 to the key from Step 1 to create a new key. For example:

    `HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Development Kit\1.2`

    or…

    `HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Runtime Environment\1.2`

4. Read the value of the `JavaHome` variable for the key from Step 3 to find the location of the JDK or JRE.

If you launch the `DBMON` debugging tool and then restart your web server, ServletExec/ISAPI displays the Java VM settings in the `DBMON` console window during initialization. You can examine the classpath displayed by ServletExec/ISAPI in the `DBMON` console window to see which version of the JDK is being used.

See Section 10.4 of the ServletExec 4.1 User Guide for a discussion of `DBMON`.

2.3 Uninstalling Other Servlet Engines

Very Important

You **must** uninstall any other servlet engines previously installed for use with Microsoft IIS before installing and using ServletExec/ISAPI. In particular, the Filter DLL registry entries or ISAPI Filters metabase entries associated with other servlet engines must be removed

See Section 2.5.4 for more information.

2.4 Running the ServletExec/ISAPI Installer

Double-click the ServletExec/ISAPI installer icon to launch Setup. Please review the ServletExec License Agreement carefully and check the READ ME displayed by the setup program for any last minute notices before continuing.

2.5 What Was Installed and/or Modified?

When you installed ServletExec/ISAPI, three changes were made to your system:

- The `ServletExec ISAPI` directory was created.
- The `ServletExec_ISAPI.dll` file was installed.
- Registry and/or metabase entries for ServletExec/ISAPI were created or modified.

The following sections describe each of these changes.

2.5.1 The ServletExec ISAPI Directory

The `ServletExec ISAPI` directory was created in the location you selected during the installation process. The default location suggested by the installer is:

    C:\Program Files\New Atlanta\ServletExec ISAPI

There are no restrictions on the location of the `ServletExec ISAPI` directory.

Important

Do not move the `ServletExec ISAPI` directory after installation. There is a registry entry that allows ServletExec/ISAPI to find this directory. If you move this directory, ServletExec/ISAPI will not be able to find its configuration files.

See Section 2.5.4 for more information.

The `ServletExec ISAPI` directory contains the following subdirectories:

`classes`	The `classes` subdirectory is automatically added to the ServletExec VM classpath. Place class files that will be shared by multiple servlets in this directory.
	See the READ ME file within the `classes` subdirectory for more information.
`Documentation`	This subdirectory contains the ServletExec User Guide, and the Servlet 2.3 and JSP 1.2 API documentation from Sun Microsystems.
`Examples`	This subdirectory contains examples of a web application and a JSP page with an embedded bean.
	See the appropriate chapters of the ServletExec User Guide for instructions on running these examples.

`lib`	The `lib` subdirectory contains the Java archive (.jar) files required by ServletExec/ISAPI.

Important

Do not modify the contents of the `lib` directory!

`Servlet Logs`	This subdirectory contains the `Servlet.log` files that capture the output from the servlet `log()` method. In some cases, this directory isn't created until the first ServletExec initialization.
`ServletExecData`	This subdirectory contains the ServletExec/ISAPI configuration files. These files are generally modified using the ServletExec Admin UI, but can also be edited manually. *See the ServletExec User Guide for more information on the Admin UI.*
`Servlets`	This subdirectory contains the class files for servlets hosted by ServletExec/ISAPI. *See the READ ME in the `Servlets` directory and the ServletExec User Guide for more information.*
`webapps`	One of these subdirectories exists for each virtual server configured. It contains the "default" folder for automatically deploying web applications hosted by ServletExec/ISAPI. *See the READ ME in the `webapps` directory and the ServletExec User Guide for more information.*

2.5.2 NTFS Permissions

Important

If you're using the NT File System (NTFS), permissions for the `ServletExec ISAPI` directory and its subdirectories **must** be set so that ServletExec/ISAPI has read and write access to these directories. Because ServletExec/ISAPI runs as part of the IIS process, it will run as different users at different times.

The following Groups should be granted Full Control to the `ServletExec ISAPI` directory and its subdirectories: `SYSTEM` and `Authenticated Users`.

The following User should be granted Full Control to the `ServletExec ISAPI` directory and its subdirectories: `IUSR_<server-name>` (the user created by IIS for processing requests for anonymous users).
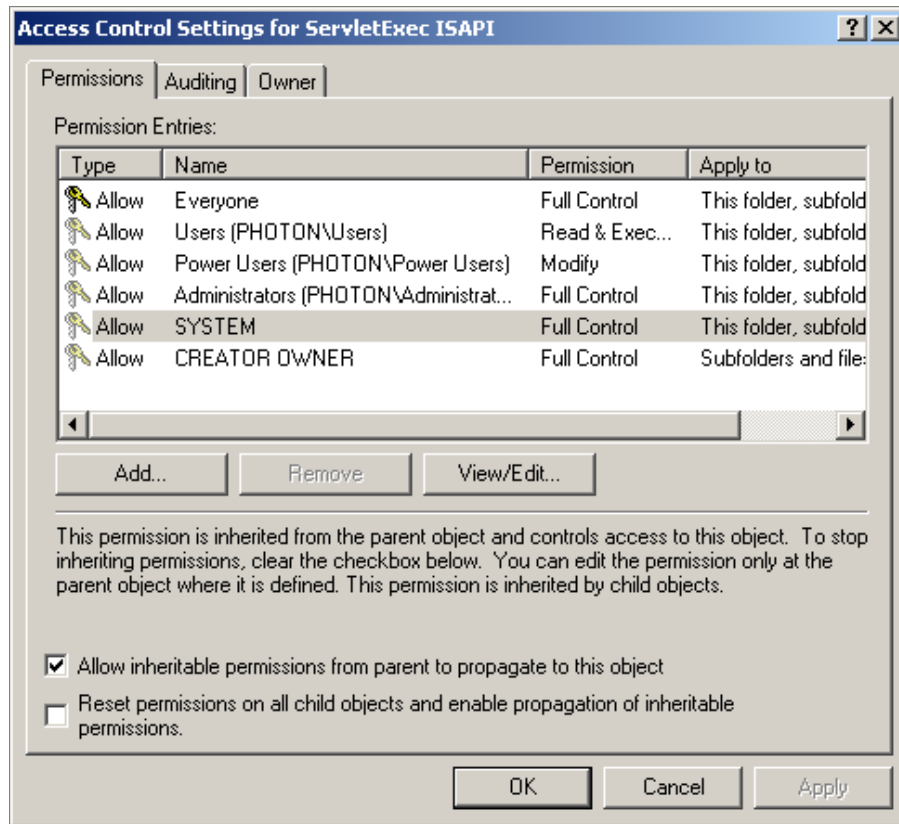
Figure 1. ISAPI NTFS Permission Settings (Windows 2000)

2.5.3 ServletExec_ISAPI.dll

The `ServletExec_ISAPI.dll` dynamic link library (DLL) was installed in the directory you selected during the installation process. This directory **must** be mapped to a Microsoft IIS virtual directory and the virtual directory **must** have `execute` permission enabled.

In the default IIS installation, the `SCRIPTS` virtual directory is configured and mapped to the `C:\InetPub\Scripts` physical directory. If the ServletExec/ISAPI installer found this default configuration, it copied `ServletExec_ISAPI.dll` to the `C:\InetPub\Scripts` directory and did not prompt you for a location for this file.

To confirm your settings for IIS

1. In Internet Service Manager, open the Properties dialog for the virtual directory that maps to the physical directory in which `ServletExec_ISAPI.dll` resides. (By default, this is the `SCRIPTS` virtual directory).

2. Verify that the **Name** parameter under **Application Settings** is available and that execute permission is enabled. If it is not available, click **Create** (IIS 4.0 only).

3. Click **Apply**, and click **OK**.

2.5.4 Registry & Metabase Entries

The ServletExec/ISAPI installer creates a new registry entry with the following key:

    HKEY_LOCAL_MACHINE\SOFTWARE\New Atlanta Communications\ServletExec ISAPI

This key contains a single parameter named `Home`, that contains the path to the `ServletExec ISAPI` directory.

Note

If you move the `ServletExec ISAPI` directory after installation, you must modify this key to contain the new path.

2.5.4.1 Filter DLLs Registry Entry

Some servlet engines do not use the metabase ISAPI Filter entry for Microsoft IIS (as described in the following section), but instead continue to use the old `Filter DLLs` registry entry. This parameter has the following key:

    HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters

Very Important

You **must** remove entries for other servlet engines previously installed for Microsoft IIS from the `Filter DLLs` parameter. The uninstallers for most servlet engines do not automatically remove this entry.

2.5.4.2 Metabase ISAPI Filter Entry

The ServletExec/ISAPI installer automatically modifies the IIS 4.0 (Windows NT 4.0) or IIS 5.0 (Windows 2000) metabase to add the ISAPI Filter entry.

Very Important

You **must** remove entries for other servlet engines you may have previously installed. The uninstallers for most servlet engines do not automatically remove the metabase ISAPI Filter entry.

To examine or manually modify the metabase ISAPI Filter under IIS 5.0

1. Click **Start**, point to **Settings** and click **Control Panel**.
2. In Control Panel, double-click **Administrative Tools**.
3. In Administrative Tools, double-click **Internet Services Manager**.
4. Expand the **Internet Services Manager** entry until you can see the icon for your server.
5. Right-click the server icon, and then click **Properties**. You should see a dialog similar to Figure 2. Make sure you've opened the **Properties** dialog for the server as illustrated in Figure 2, and not for the **Default Web Site** (or any other web site you may have configured).
6. In **Properties**, check to make sure **WWW Service** is selected in the **Master Propertie**s list (*see* Figure 2), and then click **Edit**.
7. In **Master Properties**, click the **ISAPI Filters** tab (*see* Figure 3).
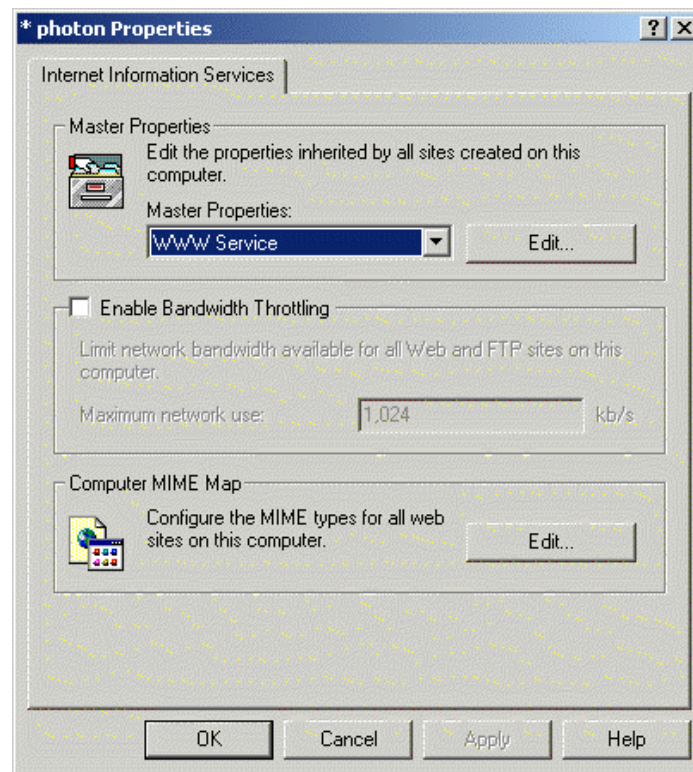
Figure 2. Server Properties (IIS 5.0)
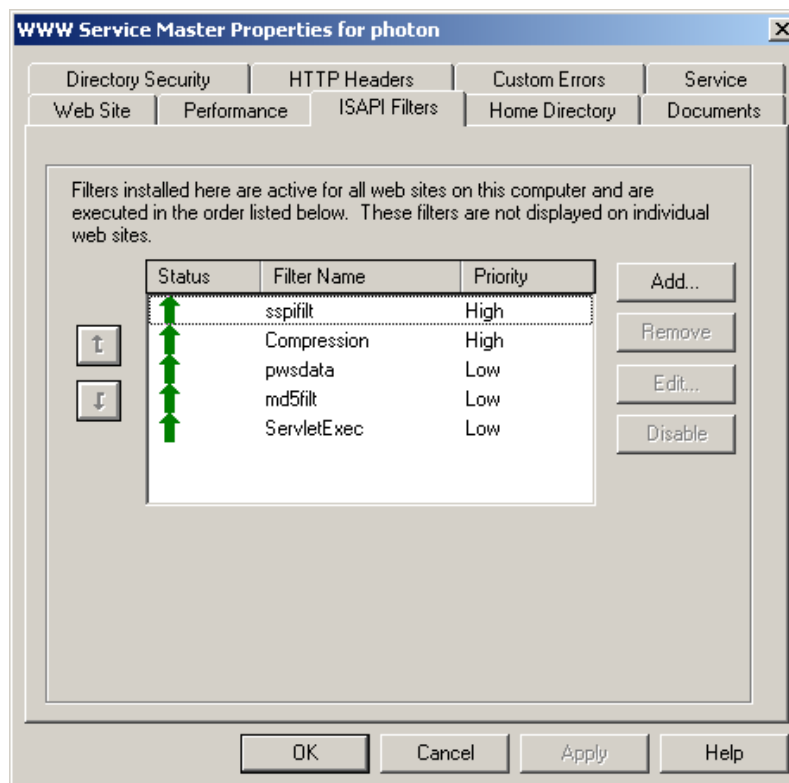
Figure 3. ISAPI Filters (IIS 5.0)

8. Click any previously installed servlet engines, and then click **Remove**.
9. Click **ServletExec,** and then click **Edit** to examine or modify it (if you wish). If you do not see **ServletExec**, click **Add** and follow the instructions to add the ServletExec/ISAPI entry, making sure to point to `ServletExec_AS.dll`.

To examine or manually modify the metabase ISAPI Filter under IIS 4.0

1. Open Internet Service Manager (Microsoft Management Console) and expand the **Internet Information Server** entry until you can see the icon for your server.
2. Right-click the server icon, and then click **Properties**. You should see a dialog similar to Figure 4. Make sure you've opened the **Properties** dialog for the server as illustrated in Figure 4, and not for the **Default Web Site** (or any other web site you may have configured).
3. In **Properties**, check to make sure **WWW Service** is selected in the **Master Propertie**s list (*see* Figure 4), and then click **Edit**.
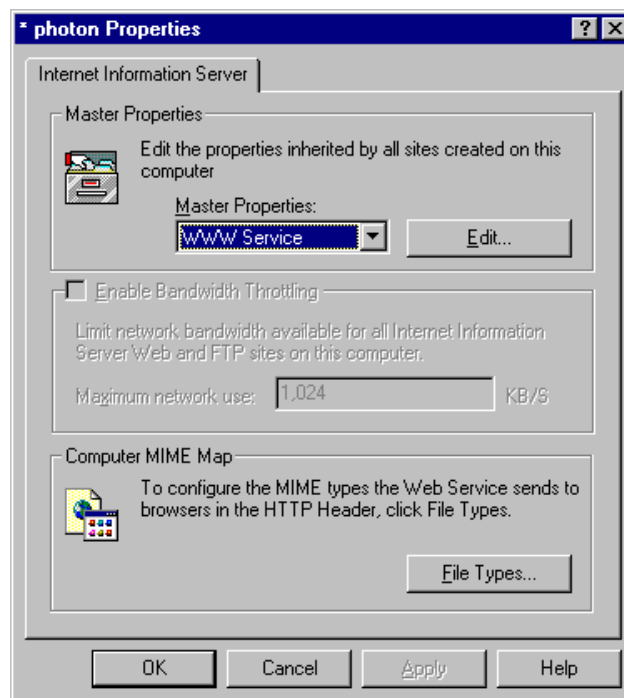
Figure 4. Server Properties (IIS 4.0)

4. In **Master Properties**, click the **ISAPI Filters** tab (*see* Figure 5).
5. Click any previously installed servlet engines, and then click **Remove**.
6. Click **ServletExec,** and then click **Edit** to examine or modify it (if you wish). If you do not see **ServletExec**, click **Add** and follow the instructions to add the ServletExec/ISAPI entry, making sure to point to `ServletExec_AS.dll`.
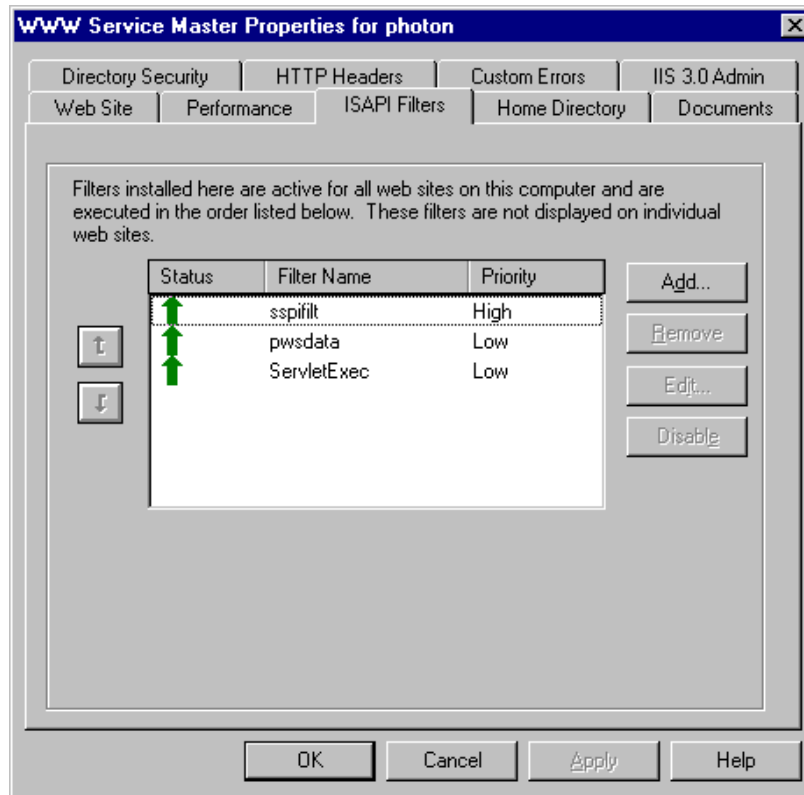
Figure 5. ISAPI Filters

2.6 ServletExec Admin Username & Password

2.6.1 Windows NT 4.0 and Windows 2000

The username and password assigned in the ServletExec Admin User Interface must match those of an existing Windows user as defined in the User Manager.

See Section 1.4.1 of the ServletExec 4.1 User Guide for information on setting the username and password for the ServletExec Admin UI.

2.6.2 IIS 4.0 (Windows NT 4.0) and IIS 5.0 (Windows 2000)

Basic Authentication must be enabled for the ServletExec Admin Username and Password to work properly. Basic Authentication is disabled by default by IIS, so you must enable it. Basic Authentication must be enabled in the WWW Service Master Properties and the individual web server Properties dialogs.

To enable Basic Authentication

1. Click **Start**, point to **Settings** and click **Control Panel**.
2. In Control Panel, double-click **Administrative Tools**.
3. In Administrative Tools, double-click **Internet Services Manager**.
4. Expand the **Internet Services Manager** entry until you can see the icon for your server.

5. Right-click the server icon, and then click **Properties**. Make sure you've opened the **Properties** dialog for the server as illustrated in Figure 4, and not for the **Default Web Site** (or any other web site you may have configured).
6. In **Properties**, check to make sure **WWW Service** is selected in the **Master Propertie**s list, and then click **Edit**.
7. Click the **Directory Security** tab and under **Anonymous Access and Authentication Control**, click **Edit**. You should then see a dialog similar to Figure 6.
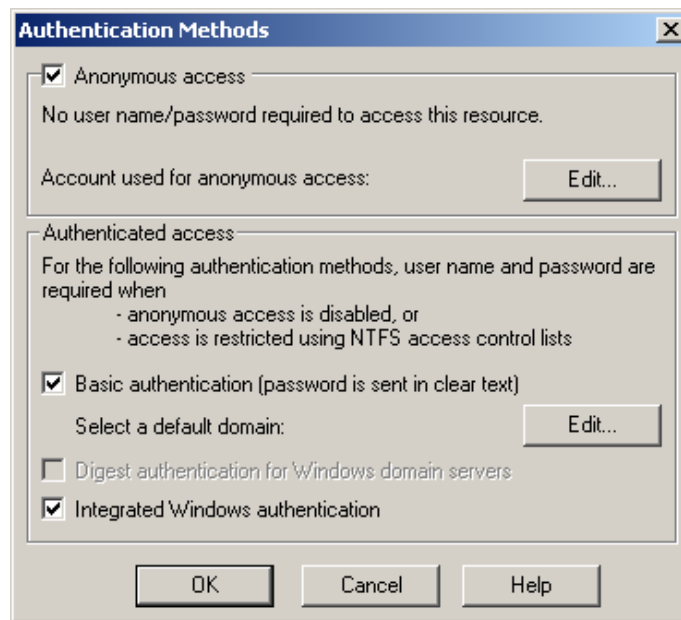
Figure 6. Authentication Methods

8. Check to make sure **Basic Authentication** is selected. By default, IIS authenticates users only in the local domain. If you need to authenticate users in a network domain, beside **Default domain for basic Authentication:**, click **Edit** and specify the network domain.
9. Click **OK** to close each dialog.

2.7 User Accounts for Microsoft IIS

Because ServletExec/ISAPI runs as part of the Microsoft IIS process, your servlets will run under different user accounts at different times. This will primarily affect their ability to read and write to the file system because access to the NT File System (NTFS) is based on the user account of the process.

Rules Governing IIS User Accounts

1. During normal request processing in your servlet's `service()`, `doGet()`, or `doPost()` method, your servlet will be running under the user account of the authenticated user, if the user had to enter a username and password to access

your servlet. Otherwise, your servlet will be running under the `IUSR_<server-name>` account (the account under which IIS runs anonymous user requests).

2. If your servlet is configured to be loaded by ServletExec/ISAPI during initialization, your `init()` method will be executed under the `SYSTEM` account. Otherwise, if your servlet is loaded when it receives its first request, the rules for item #1, above, apply.

3. Normally, your servlet's `destroy()` method will run under the SYSTEM account when it is invoked due to a shutdown of ServletExec/ISAPI. However, if your servlet is reloaded due to a class file modification, the rules for item #1, above, apply.

Make sure the appropriate users are given NTFS read and/or write permission for any directories for which your servlet needs to read and/or write files.

2.8 Reinitializing ServletExec/ISAPI

To reinitialize ServletExec/ISAPI (which must be done after modifying the ServletExec/ISAPI classpath, for example), you must completely stop and restart IIS.

To reinitialize IIS 4.0 on Windows NT 4.0

You must stop and then restart the **IIS Admin Service**. There are two ways to do this in Windows NT 4.0.

Using the Services control panel…

1. In Control Panel, click Services.
2. Click **IIS Admin Service**, and then click **Stop**.
3. Restart IIS.

Using the command line…

- Execute the batch file `stop_iis.bat` (found in the `ServletExec ISAPI` directory).

To reinitialize IIS 5.0 on Windows 2000

There are three ways to do this in Windows 2000.

Using the Internet Services Manager administrative tool…

- Just click **Restart IIS**.

Using the Services control panel…

1. In Control Panel, click **Services**.
2. Click **IIS Admin Service**, and then click **Stop**.
3. Restart IIS.

Using the command line…

- Execute the batch file `stop_iis.bat` (found in the `ServletExec ISAPI` directory).

2.9 Uninstalling ServletExec/ISAPI

To completely uninstall ServletExec/ISAPI

1. Stop **IIS** as described in Section 2.8.
2. Close any open DBMON windows.
3. Open **Control Panel** and click **Add/Remove Programs**.
4. In the list, click **ServletExec 4.1 ISAPI**, and then click **Add/Remove** or **Change/Remove**, depending on your operating system.
5. Sometimes the ServletExec/ISAPI uninstaller may not be able to remove all of the installed files. Check to see if the uninstaller deleted the `ServletExec ISAPI` directory, and if not, delete it.

3

3. ServletExec/NSAPI: Netscape/iPlanet Plugin (Windows)

Installing the Windows 2000/Windows NT plugin servlet engine for Netscape Enterprise Server and iPlanet Web Server

S ervletExec/NSAPI is a plugin servlet engine for Netscape Enterprise Server (NES) and iPlanet Web Server (iWS). ServletExec/NSAPI enables you to deploy Java Servlets and JavaServer Pages (JSP) on NES/iWS in a standard, robust, high-performance environment.

The ServletExec/NSAPI 4.1 installer for Windows is named `ServletExec_NSAPI_41.exe`. If you have not already done so, you can download this installer from:

> **http://www.newatlanta.com/downloads/index.jsp**

This chapter contains important information that will allow you to verify your installation of ServletExec/NSAPI. It will also be useful if you decide to uninstall ServletExec/NSAPI and need to make sure you've completely removed all installed components.

Completing the ServletExec/NSAPI installation requires modifications to the NES/iWS `obj.conf` configuration file. The ServletExec/NSAPI installer gives you the option to allow it to automatically update the `obj.conf` file. Instructions in this chapter allow you to verify these changes, or to manually edit `obj.conf` if you chose not to let the installer do so automatically.

If you're using iWS 6.0 or higher, you'll also need to modify the `magnus.conf` file. These modifications are covered in detail in this chapter.

3.1 Upgrading from a Previous Version

You can upgrade to ServletExec 4.1 from all previous versions except versions 3.0C or 3.0E. If you have either of these versions installed, you will need to completely uninstall that version and do a clean installation of ServletExec 4.1.

To upgrade to a new version of ServletExec/NSAPI and maintain your old configuration settings

1. Stop your web server. If you've installed ServletExec/NSAPI for multiple NES/iWS server instances, stop each one that is running.
2. Close all open applications, making sure to include DBMON.
3. Make backup copies of the `ServletExecData`, `Servlets`, `classes` (if you've added an additional one), and `webapps` (versions 3.1 and above only) subdirectories for all NES/iWS server instances for which you've installed ServletExec/NSAPI. The location of this directory varies depending upon your version of ServletExec.

   ▪ For ServletExec/NSAPI 2.2 and earlier, the default locations are within:

      `C:\Netscape\SuiteSpot\Plugins\ServletExecNSAPI\https-<server-name>`

   ▪ For ServletExec/NSAPI 3.0 and later, the default locations are within:

      `C:\Program Files\New Atlanta\ServletExecNSAPI\https-<server-name>`

4. Uninstall each of the old versions of ServletExec/NSAPI for all NES/iWS server instances on which they were installed. In Control Panel, click **Add/Remove Programs** and follow the instructions on the **Install/Uninstall** tab. For ServletExec/NSAPI versions 2.2 and earlier, the uninstaller will not delete the `ServletExecNSAPI` directory and its subdirectories. You may delete these directories *after making backup copies*.
5. Run the installer for the new version of ServletExec/NSAPI, installing into the same directory as the previous installation. You will need to re-run the installer for each NES/iWS server instance for which you want to install ServletExec/NSAPI.

Note

The default installation directory for versions 2.2 and earlier is different.

Important

For a successful upgrade, you must install everything into the same directory as the previous version.

6. Copy the `ServletExecData` and `Servlets` directories to the new `ServletExecNSAPI` directory created by the installer in Step 5.
7. Restart your web server.

After restarting your web server, ServletExec/NSAPI will run using your old configuration settings.

Tip

If you have any problems, you can restore the subdirectories from the backups you made in Step 3.

3.2 System Requirements

3.2.1 Operating System and Web Server

- ServletExec/NSAPI 4.1 is not supported on Windows 95, Windows 98 or Windows ME.
- ServletExec/NSAPI 4.1 supports the following operating systems and web servers:

OS	Web Server
Windows 2000, Windows NT 4.0 Server, and Windows NT 4.0 Workstation	- Netscape Enterprise Server (NES) 3.5.1 and higher - iPlanet Web Server (iWS) 4.0 and higher

Table 3. NSAPI Supported Operating Systems and Web Servers

3.2.2 JDK or JRE

ServletExec/NSAPI 4.1 is not supported on JDK or JRE 1.1.x or earlier.

To install ServletExec/NSAPI 4.1, you must first install one of these Java Development Kit (JDK™) or Java™ Runtime Environment (JRE) choices:

- JDK/JRE 1.2.2 or 1.3.1 for Windows from Sun Microsystems
- JDK/JRE for Windows 1.3 from IBM

Important

The IBM JRE does not contain a Java compiler, which is required by ServletExec to support JavaServer Pages (JSP). Be sure to install the JDK, and not the JRE, if you plan to use JSP.

You can download the Sun JDKs for Windows from:

http://java.sun.com/j2se/

You can download the IBM JDKs for Windows from:

http://www.ibm.com/developerworks/java/

As of this writing, the latest production versions of the Sun JDKs for Windows are 1.2.2 and 1.3.1. The latest production version of the IBM JDK for Windows is 1.3. These are the versions we tested and recommend using with ServletExec/NSAPI 4.1.

3.2.3 JDK/JRE Installation

To install ServletExec/NSAPI, you must have first installed a compatible version of the JDK or JRE as explained in Section 3.2.2.

Important

The JDK or JRE must be installed on a local drive and not on a mapped network drive. If the JDK/JRE has been installed on a mapped network drive, ServletExec/ISAPI will not be able to load and initialize the Java VM.

3.2.3.1 Multiple JDK/JRE Versions Installed

You may have multiple versions of the JDK/JRE installed on your system. ServletExec/NSAPI (including the installer) uses registry entries to determine which installed JDK/JRE to use. It will look for an installed JDK first and if it doesn't find one it will look for a JRE. It will look for JDK/JREs from vendors in the following order: Sun Microsystems, and then IBM. Study the complete algorithm that follows.

How ServletExec selects an installed JDK/JRE

1. Look for a Sun JDK using the following registry key:

   `HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Development Kit`

   ▪ If Sun JDK found, go to Step 2. Otherwise, look for a Sun JRE using the following key:

   `HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Runtime Environment`

   ▪ If Sun JRE found, go to Step 2. Otherwise, look for an IBM JDK or JRE using the following keys .

   `HKEY_LOCAL_MACHINE\SOFTWARE\IBM\Java Development Kit`

   or…

   `HKEY_LOCAL_MACHINE\SOFTWARE\IBM\Java Runtime Environment`

   ▪ If no Sun or IBM JDK/JRE found using the above registry keys, then abort ServletExec installer.

2. Read the `CurrentVersion` variable from the key found in Step 1. Currently, the only valid values for this variable start with "1.2" or "1.3" (including, for example "1.2.2").

3. Append the value of the `CurrentVersion` variable from Step 2 to the key from Step 1 to create a new key. For example:

   `HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Development Kit\1.2`

   or…

   `HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Runtime Environment\1.2`

4. Read the value of the `JavaHome` variable for the key from Step 3 to find the location of the JDK or JRE.

If you launch the `DBMON` debugging tool and then restart your web server, ServletExec/NSAPI displays the Java VM settings in the `DBMON` console window during initialization. You can examine the classpath displayed by ServletExec/NSAPI in the `DBMON` console window to see which version of the JDK is being used.

See Section 10.4 of the ServletExec 4.1 User Guide for a discussion of `DBMON`*.*

3.3 Uninstalling Other Servlet Engines

Very Important

You **must** uninstall any other servlet engines previously installed for use with NES/iWS before installing and using ServletExec/NSAPI. In particular, modifications to the `obj.conf` configuration file for other servlet engines must be removed. In addition, you **must** disable the NES/iWS built-in servlet support as described in the following sections **before** installing ServletExec/NSAPI.

3.3.1 Deactivate the Java Interpreter (NES 3.5.1 and higher)

For NES 3.5.1 and higher, the built-in Java interpreter is deactivated by default. If, however, it has been activated on your server you must deactivate it manually before installing ServletExec/NSAPI.

To manually deactivate the Java Interpreter

1. From the Netscape Administration Server home page, click the appropriate server.

    See the Netscape Administration Server documentation if you need help with this step.

2. Click **Programs** from the menu bar in the upper frame of the server administration page.
3. In the menu in the left frame of the Programs page, click **Java**.
4. Turn off the Java interpreter.

Tip

If you still have problems running ServletExec/NSAPI after deactivating the Java interpreter, check the `Init` directives in your `obj.conf` file as described in Section 3.6.3.

3.3.2 Disable the Built-In Servlet Engine (iWS 4.0 and higher)

For iWS 4.0 and higher, the built-in servlet engine is enabled by default. You **must** disable the built-in servlet engine **before** installing ServletExec/NSAPI.

To disable the Servlet Engine

1. From the iPlanet Administration Server home page, click the appropriate server.
2. On the Server Manager page, click the **Servlets** tab.
3. On the Enable Servlets page, click **No** under **Activate the Servlet Engine?,** and click **OK**.
4. On the resulting page, click **Save and Apply**.

3.4 Installing ServletExec/NSAPI for Multiple Servers

If you've created multiple NES/iWS server instances, you must run the ServletExec/NSAPI installer separately for each server instance for which you want to install ServletExec/NSAPI. Pay particular attention to the following items when installing ServletExec/NSAPI for multiple servers:

- Stop all servers before running the ServletExec/NSAPI installer.
- Use the same installation directory for all ServletExec/NSAPI installations.

The ServletExec/NSAPI installer will create a separate subdirectory within the `ServletExecNSAPI` directory for each server instance

See 3.6.1 for a detailed explanation of the NSAPI directory.

When installed for multiple NES/iWS server instances, each instance of ServletExec/NSAPI creates its own Java VM. Therefore, the servlets being hosted by ServletExec/NSAPI for each server instance are isolated from the others in separate VMs.

Note

ServletExec/NSAPI 3.0 and earlier required you to purchase a unique ServletExec serial number (license key) for each NES/iWS server instance on which ServletExec/NSAPI was installed. ServletExec/NSAPI licensing changed starting with 3.1 so that you now need to purchase only a single license key per physical server machine, regardless of how many NES/iWS server instances or ServletExec/NSAPI instances are installed on the machine.

3.5 Running the ServletExec/NSAPI Installer

To run the ServletExec/NSAPI Installer

1. Double-click the ServletExec/NSAPI installer icon to launch Setup.
2. Please review the ServletExec License Agreement carefully and check the READ ME displayed by the setup program for any last minute notices before continuing. The setup program will prompt you to select an NES/iWS server instance on which to install ServletExec/NSAPI.
3. Continue following the instructions and prompts until you see confirmation of a successful installation.

3.6 What Was Installed and/or Modified?

When you installed ServletExec/NSAPI, these changes were made to your system:

- The `ServletExecNSAPI` directory was created
- Registry entries for ServletExec/NSAPI were created or modified
- The Netscape `obj.conf` configuration file was updated (if you chose to let the ServletExec/NSAPI installer modify this file; otherwise, you must manually edit `obj.conf` to support ServletExec/NSAPI)

The following sections describe each of these changes.

3.6.1 The ServletExec NSAPI Directory

The `ServletExecNSAPI` directory was created in the location you selected during the installation process. The default location suggested by the installer is:

    C:\Program Files\New Atlanta\ServletExecNSAPI

There are no restrictions on the location of the `ServletExecNSAPI` directory.

Important

Do not move the `ServletExecNSAPI` directory after installation. There is a registry entry that allows ServletExec/NSAPI to find this directory. If you move this directory, ServletExec/NSAPI will be unable to find its configuration files.

See 3.6.2 for a detailed discussion of registry entries.

The `ServletExecNSAPI` directory contains the following subdirectories:

`Documentation`	This subdirectory contains the ServletExec User Guide, and the Servlet 2.3 and JSP 1.2 API documentation from Sun Microsystems.
`Examples`	This subdirectory contains examples of a web application and a JSP page with an embedded bean. *See the appropriate chapters of the ServletExec User Guide for instructions on running these examples.*
`ExampleServlets`	The `ExampleServlets` subdirectory contains example servlets that are copied to the `Servlets` subdirectory for each NES/iWS server instance on which ServletExec/NSAPI is installed. The example servlets are here just for use by the installer.
`https-<server-name>`	There is one `https-<server-name>` subdirectory for each NES/iWS server instance on which ServletExec/NSAPI is installed. *See further discussion of this directory directly below.*
`installerFiles`	The `installerFiles` subdirectory contains files use by the ServletExec/NSAPI setup program for installing ServletExec/NSAPI on multiple NES/iWS server instances. This directory and its contents are only used by the installer.
`lib`	The `lib` sub-directory contains the Java archive (.jar) files required by ServletExec/NSAPI.

Important

Do not modify the contents of this directory!

The `https-<server-name>` subdirectories of the `ServletExecNSAPI` directory each contain the following subdirectories.

`classes`	The `classes` subdirectory is automatically added to the ServletExec VM classpath. Place class files that will be

shared by multiple servlets in this directory.

See the READ ME file within the `classes` subdirectory for more information.

`Servlet Logs`	This subdirectory contains the `Servlet.log` files that capture the output from the servlet `log()` method. In some cases, this directory doesn't get created until the first ServletExec initialization.
`ServletExecData`	This subdirectory contains the ServletExec/NSAPI configuration files. These files are generally modified using the ServletExec Admin UI, but can also be edited manually.
`Servlets`	This subdirectory contains the class files for servlets hosted by ServletExec/NSAPI. *See the READ ME in the `Servlets` directory and the ServletExec User Guide for more information.*
`webapps`	One of these subdirectories exists for each virtual server configured. This subdirectory contains the "default" folder for automatically deploying web applications hosted by ServletExec/NSAPI. *See the READ ME in the `webapps` directory and the ServletExec User Guide for more information.*

3.6.2 Registry Entries

The ServletExec/NSAPI installer creates a new registry entry for each web server instance for which it is installed with the following key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\New Atlanta
Communications\ServletExecNSAPI\https-<server-name>
```

This key contains a single parameter named `Home`, that contains the path to the web server's subdirectory within the `ServletExecNSAPI` directory.

Note

If you move the `ServletExecNSAPI` directory after installation, you must modify the key for each server to contain the new path.

3.6.3 NES/iWS Configuration File (`obj.conf`) Modifications

The NES/iWS server implementation requires that the `obj.conf` configuration file be modified to enable installation of NSAPI plugins such as ServletExec/NSAPI. This section describes the modifications performed by the ServletExec/NSAPI installer.

Important

If you chose not to allow the installer to make these modifications, you must make them manually.

Several lines must be added to the `obj.conf` configuration file for each server for which ServletExec/NSAPI is installed. <u>The location of these lines within the `obj.conf` file is very important</u>.

1. Add the following lines to the beginning of `obj.conf` before the other `Init` directives:

   ```
   Init fn="load-modules" shlib="<path>/ServletExec_NSAPI.dll"
       funcs="ServletExecInit,ServletExecFilter,ServletExecService"
   Init fn="ServletExecInit"
   ```

   where `<path>` is the full path to the `ServletExecNSAPI` directory.

   Note

   The first `Init` directive will normally appear on a single line within the `obj.conf` file. It's shown as spanning two lines here for formatting reasons. It may span two lines within `obj.conf`, in which case the second line must begin with a tab or space character).

   Important

   If you had previously activated, then deactivated the Java Interpreter for NES 3.5.1 or higher, two `Init` directives similar to the following will still be in your `obj.conf` file:

   ```
   Init funcs="SJavaBootInit" shlib=".." fn="load-modules"
   ```

   ```
   Init classpath=".." ldpath=".." fn="SJavaBootInit"
   ```

   You must either: (a) make sure the ServletExec/NSAPI `Init` directives appear before these two Java Interpreter directives; or, (b) remove the Java Interpreter directives.

   Important—iWS 6.0 or Higher Users

   If you're using iWS 6.0 or higher, you must add the `Init` directives to the `magnus.conf` file rather than the `obj.conf` file.

2. The following lines must be added within the `<Object name=default>` directives:

   ```
   NameTrans fn="ServletExecFilter"
   Service method="(GET|HEAD|POST)" type="magnus-internal/nac"
           fn="ServletExecService"
   ```

   Note

   The `Service` directive will normally appear on a single line within the `obj.conf` file. It's shown as spanning two lines here for formatting reasons. It may span two lines within `obj.conf`, in which case the second line must begin with a tab or space character.

Important

The `NameTrans` directive for ServletExec/NSAPI must appear first in the list of `NameTrans` directives for the `default` object. The `Service` directive for ServletExec/NSAPI must similarly appear first in the list of `Service` directives for the `default` object.

Figure 7 shows a partial `obj.conf` file for NES 3.6 with ServletExec/NSAPI installed. The default installation locations were used for both the server and ServletExec/NSAPI. The changes made for ServletExec/NSAPI are highlighted.

3.6.4 VerifyObjConf.bat

Within the `https-<server-name>` subdirectory of the `ServletExecNSAPI` directory is the command file `VerifyObjConf.bat`. This command file can be executed to automatically check your `obj.conf` file for errors. `VerifyObjConf.bat` creates a file named `Verify.log` in the `https-<server-name>` subdirectory that contains a list of warnings and errors.

3.7 Uninstalling ServletExec/NSAPI

To uninstall ServletExec/NSAPI

1. In Control Panel, click **Add/Remove Programs**.
2. Click **ServletExec 4.1 NSAPI**, and then click **Add/Remove** or **Change/Remove**, depending on your operating system.
3. When prompted by the uninstaller, select one of the following, depending on the scenario:

   - **Uninstall a ServletExec instance** to uninstall ServletExec/NSAPI for a single NES/iWS instance

   - **Completely Uninstall ServletExec** to uninstall ServletExec/NSAPI for all NES/iWS web server instances

4. The uninstaller may not be able to remove all installed files. Check for the `https-<server-name>` directory from within the `ServletExecNSAPI` directory, and delete it if it remains. The default location of the `ServletExecNSAPI` directory is within the `C:\Program Files\New Atlanta` directory.
5. Edit the `obj.conf` configuration file for the NES/iWS server instance(s) and remove the lines that were added for ServletExec/NSAPI (as described in Section 3.6.3)

Important

The installer will not automatically remove these lines. If you don't manually remove them, the web server will not run.

---

6. If you chose to completely uninstall ServletExec/NSAPI from your system, delete the `ServletExecNSAPI` directory.

See the following example in Figure 7

```
Init fn="load-modules"
    shlib="C:/Program Files/New Atlanta/ServletExec NSAPI/ServletExec_NSAPI.dll"
    funcs="ServletExecInit,ServletExecFilter,ServletExecService"
Init fn="ServletExecInit"
Init fn=flex-init access="C:/Netscape/SuiteSpot/https-ntserver1/logs/access" format.access="%Ses->client.ip% -
%Req->vars.auth-user% [%SYSDATE%] \
            "%Req->reqpb.clf-request% \
            " %Req->srvhdrs.clf-status% %Req->srvhdrs.content-length%"
Init fn=load-types mime-types=mime.types
Init fn="load-modules"
    funcs="CM_Init,CM_Delete,CM_Index,CM_Get,CM_Put,CM_Move,CM_MkDir,CM_Post,CM_Edit,
CM_Unedit,CM_Save,CM_Lock,CM_Unlock,CM_RevLabel,CM_RevLog,CM_SetAttr,CM_GetAttr,
CM_GetAttrNames,CM_RevAdd,CM_RevNum,CM_Copy,CM_GetPS,CM_StartRev,CM_StopRev"
    shlib="C:/Netscape/SuiteSpot/plugins/content_mgr/bin/content_mgr.dll"
    NativeThread="no"
Init fn="CM_Init" webconfig="C:/Netscape/SuiteSpot/https-ntserver1/config/webpub.conf"
Init fn="load-modules" funcs="es-search-init,es-search,es-search-nametrans"
    shlib="C:/Netscape/SuiteSpot/plugins/search/bin/nsir.dll" NativeThread="no"
Init fn="es-search-init"
    systemini="C:/Netscape/SuiteSpot/httpsntserver1/config/webpub.conf"
    errordb="C:/Netscape/SuiteSpot/plugins/search/admin/nsir.err"
    userdefsdb="C:/Netscape/SuiteSpot/plugins/search/admin/userdefs.ini"
Init fn="load-modules"
    funcs="ns_agentInit,agent_name_trans,ns_agentCmdHandler,ns_agentType"
    shlib="C:/Netscape/SuiteSpot/plugins/agents/bin/agents.dll" NativeThread="no"
Init fn="ns_agentInit"
    systemini="C:/Netscape/SuiteSpot/plugins/agents/data/agent_system.ini"
    uidir="C:/Netscape/SuiteSpot/plugins/agents/data" LateInit="yes"

<Object name=default>
NameTrans fn="ServletExecFilter"
NameTrans fn=pfx2dir from=/ns-icons dir="C:/Netscape/SuiteSpot/ns-icons"
NameTrans fn=pfx2dir from=/mc-icons dir="C:/Netscape/SuiteSpot/ns-icons"
NameTrans fn="pfx2dir" from="/help" dir="C:/Netscape/SuiteSpot/manual/https/ug"
NameTrans from="/Agents" fn="agent_name_trans"
NameTrans fn=pfx2dir from="/search-ui" dir="C:/Netscape/SuiteSpot/plugins/search/ui"
NameTrans fn="es-search-nametrans" from="/search"
NameTrans fn="pfx2dir" from="/webpub-ui"
          dir="C:/Netscape/SuiteSpot/plugins/content_mgr/ui"
NameTrans fn="pfx2dir" from="/publisher"
          dir="C:/Netscape/SuiteSpot/plugins/content_mgr/client"
NameTrans fn=document-root root="C:/Netscape/SuiteSpot/docs"
PathCheck fn=nt-uri-clean
PathCheck fn="check-acl" acl="default"
PathCheck fn=find-pathinfo
PathCheck fn=find-index index-names="index.html,home.html"
ObjectType fn=type-by-extension
ObjectType fn=force-type type=text/plain
Service method="(GET|HEAD|POST)" type="magnus-internal/nac" fn="ServletExecService"
Service method=(GET|HEAD) type=magnus-internal/imagemap fn=imagemap
Service method=(GET|HEAD) type=magnus-internal/directory fn=index-common
Service fn="CM_Delete" method="DELETE"
Service fn="CM_Index" method="INDEX"
Service fn="CM_Get" method="(GET|HEAD)"
Service fn="CM_Put" method="PUT"
```

Figure 7. Partial `obj.conf` **Configuration File for NES 3.6**

4. ServletExec/NSAPI: Netscape/iPlanet Plugin (SPARC Solaris)

Installing the SPARC Solaris plugin servlet engine for Netscape Enterprise Server and iPlanet Web Server

ervletExec/NSAPI 4.1 is a plugin servlet engine for Netscape Enterprise Server (NES) and iPlanet Web Server (iWS). ServletExec/NSAPI enables you to deploy Java Servlets and JavaServer Pages (JSP) on NES/iWS in a standard, robust, high-performance environment.

The ServletExec/NSAPI 4.1 installer for SPARC™ Solaris™ is named `ServletExec_NSAPI_41.sh`. If you have not already done so, you can download this installer from:

This chapter contains important information that will allow you to verify your installation of ServletExec/NSAPI. It will also be useful if you decide to uninstall ServletExec/NSAPI and need to make sure you've completely removed all installed components.

Completing the ServletExec/NSAPI installation requires modifications to the NES/iWS `obj.conf` configuration file and to the server `start` script. The ServletExec/NSAPI installer gives you the option to allow it to automatically update the `obj.conf` file and the server `start` script. Instructions in this chapter allow you to verify these changes, or to manually edit `obj.conf` and the server `start` script if you chose not to let the installer do so automatically.

If you're using iWS 6.0 or higher, you'll also need to modify the `magnus.conf` file. These modifications are covered in detail in this chapter.

4.1 Upgrading from a Previous Version

You can upgrade to ServletExec 4.1 from all previous versions except versions 3.0C or 3.0E. If you have either of these versions installed, you will need to completely uninstall that version and do a clean installation of ServletExec 4.1.

To upgrade to a new version of ServletExec/NSAPI and maintain your old configuration settings

1. Stop your web server. If you've installed ServletExec/NSAPI for multiple NES/iWS server instances, stop each one that is running.
2. Make backup copies of the `ServletExecData`, `Servlets`, `classes` (if you've added an additional one), and `webapps` (versions 3.1 and above only) directories for all NES/iWS server instances for which you've installed ServletExec/NSAPI.

    - For ServletExec/NSAPI 2.2 and earlier, the default locations are within:

        `/usr/local/netscape/suitespot/plugins/ServletExecNSAPI/https-`
        `<server-name>`

    - For ServletExec/NSAPI 3.0 and 3.1, the default locations are within:

        `/usr/local/NewAtlanta/ServletExecNSAPI/https-<server-name>`

3. Close all open applications.
4. Uninstall the old version of ServletExec/NSAPI by running the installer for that version. Choose to Remove ServletExec Completely when prompted.
5. Run the ServletExec/NSAPI 4.1 installer. You will need to re-run the installer for each NES/iWS web server instance for which you want to install ServletExec/NSAPI 4.1, installing into the same location as the previous installation.

    ### Note

    The default installation directory for versions 2.2 and earlier is different.

    ### Important

    For a successful upgrade, you must install everything into the same directory as the previous version.

6. Copy the `ServletExecData`, `Servlets`, `classes` (if you've added an additional one), and `webapps` (versions 3.1 and above only) directories to the `ServletExecNSAPI/https-<server-name>`.
7. The `obj.conf` file for each upgraded NES/iWS server instance must have the path to the `ServletExecNSAPI.so` shared library in the `Init` directive updated to include the `/bin` subdirectory. That is, in ServletExec/NSAPI versions 2.2 and earlier, the `Init` directive appears (partially) as:

        Init fn="load-modules" shlib="<path>/ServletExecNSAPI.so"

    In ServletExec/NSAPI 3.x and 4.x, the `Init` directive appears (partially) as:

        Init fn="load-modules" shlib="<path>/bin/ServletExecNSAPI.so"

where `<path>` is the full path to the `ServletExecNSAPI` directory. The default path to the `ServletExecNSAPI` directory is different in ServletExec/NSAPI 4.x than for previous versions, so you may also need to update the `<path>`.

8. The **start** script for each upgraded NES/iWS server instance may need to be updated to reflect the base location of the new ServletExec installation. For ServletExec/NSAPI 4.1 installed in the default location, the following lines should appear in the server start script:

```
ServletExecHome=/usr/local/NewAtlanta
export ServletExecHome
```

9. Restart your web server.

After restarting your web server, the new version of ServletExec/NSAPI will run using your old configuration settings.

Tip

If you have any problems, you can restore the subdirectories from the backups you made in Step 3.

4.2 System Requirements

- ServletExec/NSAPI 4.1 is not supported on JDK or JRE 1.1.x or earlier.
- ServletExec/NSAPI 4.1 supports the following operating systems and web servers…

4.2.1 Operating System and Web Server

OS	Web Server
SPARC Solaris 2.6 (with patch), SPARC Solaris 7, and SPARC Solaris 8	▪ Netscape Enterprise Server (NES) 3.5.1 and higher ▪ iPlanet Web Server (iWS) 4.0 and higher

Table 4. NSAPI/Solaris Supported Operating Systems and Web Servers

You can download the ReadMe and patch for Solaris 2.6 from:

```
ftp://sunsolve.sun.com/pub/patches/105181.readme
ftp://sunsolve.sun.com/pub/patches/105181-28.tar.Z
```

4.2.2 JDK/JRE Installation

To install ServletExec/NSAPI 4.1, you must first install one of these Java Development Kit (JDK™) or the Java™ Runtime Environment (JRE) choices:

- JDK 1.2.x (or higher) from Sun Microsystems
- JRE 1.2.x (or higher) from Sun Microsystems

Important

The JRE does not contain a Java compiler, which is required by ServletExec to support JavaServer Pages (JSP). Be sure to install the JDK, and not the JRE, if you plan to use JSP.

You can download the Sun JDKs for SPARC Solaris from:

http://java.sun.com/j2se/

Note

Sun has created two versions of the JDK for Solaris, which it refers to as the Production and Reference implementations (or releases). *We recommend using only the latest Production release of the JDK with ServletExec/NSAPI.* We do not recommend using Reference releases of the JDK in production environments.

Important–iPlanet Users

Due to a bug in either iPlanet or the JVM which causes the JVM to hang under stress, we do not recommend using JDK/JRE 1.3.x or 1.4.0 beta. Instead, we recommend using the latest production release of JDK 1.2. This hanging problem only occurs with the ServletExec/NSAPI on Solaris product.

4.2.2.1 Multiple JDK/JRE Versions Installed

You may have multiple versions of the JDK/JRE installed on your system. If the installation script detects multiple versions, it will ask you to select which one to use. Uninstalling Other Servlet Engines

Very Important

You **must** uninstall any other servlet engines previously installed for use with NES/iWS before installing and using ServletExec/NSAPI. In particular, modifications to the `obj.conf` configuration file for other servlet engines must be removed. In addition, you **must** disable the NES/iWS built-in servlet support as described in the following sections **before** installing ServletExec/NSAPI.

4.2.3 Deactivate the Java Interpreter (NES 3.5.1 and higher)

For NES 3.5.1 and higher, the built-in Java interpreter is deactivated by default. If, however, it has been activated on your server you must deactivate it manually before installing ServletExec/NSAPI.

To manually deactivate the Java Interpreter

1. From the Netscape Administration Server home page, click the appropriate server.

   See the Netscape Administration Server documentation if you need help with this step.

2. Click **Programs** from the menu bar in the upper frame of the server administration page.
3. In the menu in the left frame of the Programs page, click **Java**.
4. Turn off the Java interpreter.

4.2.4 Disable the Built-In Servlet Engine (iWS 4.0 and higher)

For iWS 4.0 and higher, the built-in servlet engine is enabled by default. You **must** disable the built-in servlet engine **before** installing ServletExec/NSAPI.

To disable the Servlet Engine

1. From the iPlanet Administration Server home page, click the appropriate server.
2. On the Server Manager page, click the **Servlets** tab.
3. On the Enable Servlets page, click **No** under **Activate the Servlet Engine?**, and click **OK**.
4. On the resulting page, click **Save and Apply**.

4.3 Installing ServletExec/NSAPI for Multiple Servers

If you've created multiple NES/iWS server instances, you must run the ServletExec/NSAPI installer separately for each server instance for which you want to install ServletExec/NSAPI. Pay particular attention to the following items when installing ServletExec/NSAPI for multiple servers:

- Stop all servers before running the ServletExec/NSAPI installer.
- Use the same installation directory for all ServletExec/NSAPI installations.

The ServletExec/NSAPI installer will create a separate subdirectory within the `ServletExecNSAPI` directory for each server instance.

See 4.5.1 for a detailed discussion of the NSAPI directory.

When installed for multiple NES/iWS server instances, each instance of ServletExec/NSAPI creates its own Java VM. Therefore, the servlets being hosted by ServletExec/NSAPI for each server instance are isolated from the others in separate VMs.

Note

3.0 and earlier versions of ServletExec/NSAPI required you to purchase a unique ServletExec serial number (license key) for each NES/iWS server instance on which ServletExec/NSAPI was installed. ServletExec/NSAPI licensing changed starting with 3.1 so that you now need to purchase only a single license key per physical server machine, regardless of how many NES/iWS server instances or ServletExec/NSAPI instances are installed on the machine.

4.4 Running the ServletExec/NSAPI Installer

The file `ServletExec_NSAPI_41.sh` is a non-graphical installation shell script.

To use the ServletExec/NSAPI Installer

1. Give the file `execute` permission.

   # chmod +x ServletExec_NSAPI_41.sh

2. From a shell command line, run it as follows:

   # ./ServletExec_NSAPI_41.sh

3. Review the ServletExec License Agreement carefully and check the README displayed by the installer for any last minute notices before continuing.
4. Continue following the prompts and instructions until you see confirmation of a successful installation.

You will be able to see a record of the completed installation steps and errors in the log file `/tmp/SEinstall.log`.

Note

The `ServletExec_NSAPI_41.sh` file is overwritten each time the installer is run. Should you have any problems with your installation, please include this log file in a message to the ServletExec Technical Support at support@newatlanta.com.

4.5 What Was Installed and/or Modified?

When you installed ServletExec/NSAPI, these changes were made to your system:

- `ServletExecNSAPI` directory was created
- Netscape Server `start` script was updated (if you chose to let the ServletExec/NSAPI installation script automatically modify this file)
- Netscape `obj.conf` configuration file was updated (if you chose to let the ServletExec/NSAPI installation script automatically modify this file)

The following sections describe each of these changes.

4.5.1 The ServletExecNSAPI Directory

The `ServletExecNSAPI` directory was created within the `/usr/local/NewAtlanta` directory, or the directory you specified during the installation process.

Important

Do not move the `ServletExecNSAPI` directory after installation. If you move this directory, ServletExec/NSAPI will be unable to find its configuration files. There are no restrictions on the location of the `ServletExecNSAPI` directory.

The `ServletExecNSAPI` directory contains the following subdirectories:

Documentation	This subdirectory contains the ServletExec User Guide, and the Servlet 2.3 and JSP 1.2 API documentation from Sun Microsystems.
Examples	This subdirectory contains examples of a web

---

application and a JSP page with an embedded bean.

See the appropriate chapters of the ServletExec User Guide for instructions on running these examples.

`https-<server-name>`	There is one `https-<server-name>` subdirectory for each NES/iWS server instance on which ServletExec/NSAPI is installed.
	See further discussion of this directory directly below.
`installerFiles`	The `installerFiles` subdirectory contains files use by the ServletExec/NSAPI setup program for installing ServletExec/NSAPI on multiple NES/iWS server instances. This directory and its contents are only used by the installer.
`lib`	The `lib` sub-directory contains the Java archive (.jar) files required by ServletExec/NSAPI.

Important

Do not modify the contents of this directory!

The `https-<server-name>` subdirectories of the `ServletExecNSAPI` directory each contain the following subdirectories.

`classes`	The `classes` subdirectory is automatically added to the ServletExec VM classpath. Place class files that will be shared by multiple servlets in this directory.
	See the README file within the `classes` subdirectory for more information.
`ServletLogs`	This subdirectory contains the `Servlet.log` files that capture the output from the servlet `log()` method. In some cases, this directory doesn't get created until the first ServletExec initialization.
`ServletExecData`	This subdirectory contains the ServletExec/NSAPI configuration files. These files are generally modified using the ServletExec Admin UI, but can also be edited manually.
`Servlets`	This subdirectory contains the class files for servlets hosted by ServletExec/NSAPI.
	See the README in the `Servlets` directory and the ServletExec User Guide for more information.
`webapps`	One of these subdirectories exists for each virtual server

configured. This subdirectory contains the "default" folder for automatically deploying web applications hosted by ServletExec/NSAPI.

See the README in the `webapps` directory and the ServletExec User Guide for more information.

4.5.2 Server `start` Script

The Netscape Server `start` script must be modified to set the `ServletExecHome`, `LD_LIBRARY_PATH`, and `LD_PRELOAD` environment variables, and possibly export the `SERVER_ROOT` environment variable. This script resides in the `https-<server-name>` subdirectory of the `ServletExecNSAPI` directory.

The following sections describe the modifications made to the server `start` script by the ServletExec/NSAPI installation script.

Important

- If you chose not to allow the installer to make these modifications, you must make them manually.

- If you choose to run a different JDK or JRE version than the one available when ServletExec/NSAPI was originally installed, you may need to modify the `start` script and change the various settings of `LD_LIBRARY_PATH` as needed.

4.5.2.1 ServletExecHome

The `ServletExecHome` environment variable must be set to the base directory of the `ServletExecNSAPI` installation directory. By default, it is "`/usr/local/NewAtlanta`". The `ServletExecHome` variable must also be exported, as shown:

```
ServletExecHome=/usr/local/NewAtlanta
export ServletExecHome
```

4.5.2.2 LD_LIBRARY_PATH

The `LD_LIBRARY_PATH` environment variable must be set to include the SPARC native threads libraries (ServletExec/NSAPI *requires* SPARC native threads). For example, if the JDK is installed in `/usr/java`, then `LD_LIBRARY_PATH` must be set as follows (for Bourne shells) in the server `start` script depending on which version of the JDK or JRE you're using.

- For JDK 1.2 and 1.3 production releases:

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/java/jre/lib/sparc
export LD_LIBRARY_PATH
```

- For JRE 1.2 and 1.3 production releases:

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/java/lib/sparc
export LD_LIBRARY_PATH
```

- For JDK 1.2 reference release:

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/java/jre/lib/sparc
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/java/jre/lib/sparc/native_thr
eads
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/java/jre/lib/sparc/classic
export LD_LIBRARY_PATH
```

- For JRE 1.2 reference release:

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/java/lib/sparc
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/java/lib/sparc/native_threads
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/java/lib/sparc/classic
export LD_LIBRARY_PATH
```

4.5.2.3 LD_PRELOAD

The `LD_PRELOAD` environment variable must be set to explicitly load `libjava.so` before any other shared objects:

```
LD_PRELOAD=libjava.so
export LD_PRELOAD
```

4.5.2.4 SERVER_ROOT

Because the Netscape Server installation directory may vary, the `SERVER_ROOT` environment variable may need to be exported to the `start` script to direct ServletExec/NSAPI to that location. This is only required if your Netscape Server installation is anywhere other than `/usr/netscape/suitespot` (which ServletExec/NSAPI will assume as the default).

If your installation uses a base installation directory other than the default, the `start` script *must* contain the following line:

```
export SERVER_ROOT
```

4.5.3 NES/iWS Configuration File (`obj.conf`)

The NES/iWS server implementation requires that the `obj.conf` configuration file be modified to enable installation of NSAPI plugins such as ServletExec/NSAPI. This section describes the modifications performed by the ServletExec/NSAPI installation script.

Important

If you chose not to allow the installer to make these modifications, you must make them manually.

Several lines must be added to the `obj.conf` configuration file for each server for which ServletExec/NSAPI is installed. The location of these lines within the `obj.conf` file is very important.

1. Add the following lines to the beginning of `obj.conf` before the other `Init` directives, where `<path>` is the full path to the `ServletExecNSAPI` directory:

```
Init fn="load-modules" shlib="<path>/bin/ServletExecNSAPI.so"
    funcs="ServletExecInit,ServletExecFilter,ServletExecService"
Init fn="ServletExecInit"
```

Note

The first `Init` directive will normally appear on a single line within the `obj.conf` file. It's shown as spanning two lines here for formatting reasons. It may span two lines within `obj.conf`, in which case the second line must begin with a tab or space character.

Important

If you had previously activated, then deactivated the Java Interpreter for NES 3.5.1 or higher, two `Init` directives similar to the following will still be in your `obj.conf` file:

```
Init funcs="SJavaBootInit" shlib=".." fn="load-modules"
Init classpath=".." ldpath=".." fn="SJavaBootInit"
```

You must either: (a) make sure the ServletExec/NSAPI `Init` directives appear before these two Java Interpreter directives; or, (b) remove the Java Interpreter directives.

Important—iWS 6.0 or Higher Users

If you're using iWS 6.0 or higher, you must add the `Init` directives to the `magnus.conf` file rather than the `obj.conf` file.

2. Add the following lines within the `<Object name=default>` directives:
```
NameTrans fn="ServletExecFilter"
Service method=("GET|HEAD|POST") type="magnus-internal/nac"
        fn="ServletExecService"
```

Important

The `NameTrans` directive for ServletExec/NSAPI must appear first in the list of `NameTrans` directives for the `default` object. The `Service` directive for ServletExec/NSAPI must similarly appear first in the list of `Service` directives for the `default` object.

Note

The `Service` directive will normally appear on a single line within the `obj.conf` file. It's shown as spanning two lines here for formatting reasons. It may span two lines within `obj.conf`, in which case the second line must begin with a tab or space character.

Figure 8 shows a partial `obj.conf` file for NES 3.6 with ServletExec/NSAPI installed. The default installation locations were used for both the server and ServletExec/NSAPI. The changes made for ServletExec/NSAPI are highlighted.

4.6 Uninstalling ServletExec/NSAPI

The ServletExec/NSAPI installation script also contains the options for uninstalling.

To uninstall ServletExec/NSAPI

1. Give the file `execute` permission.

```
# chmod +x ServletExec_NSAPI_41.sh
```

2. From a shell command line, run it as follows:

```
# ./ServletExec_NSAPI_41.sh
```

3. When prompted, select one of the following, depending on the scenario:

- **Uninstall a ServletExec instance** to uninstall ServletExec/NSAPI for a single NES/iWS instance

- **Completely Uninstall ServletExec** to uninstall ServletExec/NSAPI for all NES/iWS web server instances

If you choose to remove ServletExec/NSAPI from an NES/iWS server instance, or to remove ServletExec/NSAPI completely, the uninstaller will not remove the modifications made to either the `obj.conf` or the server `start` script.

Important

The installer will not automatically remove these lines. If you don't manually remove them, the web server will not run.

```
Init fn="load-modules"
     shlib="/usr/local/NewAtlanta/ServletExecNSAPI/bin/ServletExecNSAPI.so"
     funcs="ServletExecInit,ServletExecFilter,ServletExecService"
Init fn="ServletExecInit"
Init fn=flex-init access="/usr/local/netscape/suitespot/https-ntserver1/logs/access" format.access="%Ses-
>client.ip% - %Req->vars.auth-user% [%SYSDATE%] \
               "%Req->reqpb.clf-request% \
               " %Req->srvhdrs.clf-status% %Req->srvhdrs.content-length%"
Init fn=load-types mime-types=mime.types
Init fn="load-modules"
     funcs="CM_Init,CM_Delete,CM_Index,CM_Get,CM_Put,CM_Move,CM_MkDir,CM_Post,CM_Edit,
CM_Unedit,CM_Save,CM_Lock,CM_Unlock,CM_RevLabel,CM_RevLog,CM_SetAttr,CM_GetAttr,
CM_GetAttrNames,CM_RevAdd,CM_RevNum,CM_Copy,CM_GetPS,CM_StartRev,CM_StopRev"
     shlib="/usr/local/netscape/suitespot/plugins/content_mgr/bin/content_mgr.dll"
     NativeThread="no"
Init fn="CM_Init" webconfig="/usr/local/netscape/suitespot/https-ntserver1/config/webpub.conf"
Init fn="load-modules" funcs="es-search-init,es-search,es-search-nametrans"
     shlib="/usr/local/netscape/suitespot/plugins/search/bin/nsir.dll" NativeThread="no"
Init fn="es-search-init"
     systemini="/usr/local/netscape/suitespot/httpsntserver1/config/webpub.conf"
     errordb="/usr/local/netscape/suitespot/plugins/search/admin/nsir.err"
     userdefsdb="/usr/local/netscape/suitespot/plugins/search/admin/userdefs.ini"
Init fn="load-modules"
     funcs="ns_agentInit,agent_name_trans,ns_agentCmdHandler,ns_agentType"
     shlib="/usr/local/netscape/suitespot/plugins/agents/bin/agents.dll" NativeThread="no"
Init fn="ns_agentInit"
     systemini="/usr/local/netscape/suitespot/plugins/agents/data/agent_system.ini"
     uidir="/usr/local/netscape/suitespot/plugins/agents/data" LateInit="yes"


<Object name=default>
NameTrans fn="ServletExecFilter"
NameTrans fn=pfx2dir from=/ns-icons dir="/usr/local/netscape/suitespot/ns-icons"
NameTrans fn=pfx2dir from=/mc-icons dir="/usr/local/netscape/suitespot/ns-icons"
NameTrans fn="pfx2dir" from="/help" dir="/usr/local/netscape/suitespot/manual/https/ug"
NameTrans from="/Agents" fn="agent_name_trans"
NameTrans fn="pfx2dir" from="/search-ui" dir="/usr/local/netscape/suitespot/plugins/search/ui"
NameTrans fn="es-search-nametrans" from="/search"
NameTrans fn="pfx2dir" from="/webpub-ui"
          dir="/usr/local/netscape/suitespot/plugins/content_mgr/ui"
NameTrans fn="pfx2dir" from="/publisher"
          dir="/usr/local/netscape/suitespot/plugins/content_mgr/client"
NameTrans fn=document-root root="/usr/local/netscape/suitespot/docs"
PathCheck fn=nt-uri-clean
PathCheck fn="check-acl" acl="default"
PathCheck fn=find-pathinfo
PathCheck fn=find-index index-names="index.html,home.html"
ObjectType fn=type-by-extension
ObjectType fn=force-type type=text/plain
Service method="(GET|HEAD|POST)" type="magnus-internal/nac" fn="ServletExecService"
Service method=(GET|HEAD) type=magnus-internal/imagemap fn=imagemap
Service method=(GET|HEAD) type=magnus-internal/directory fn=index-common
Service fn="CM_Delete" method="DELETE"
Service fn="CM_Index" method="INDEX"
Service fn="CM_Get" method="(GET|HEAD)"
Service fn="CM_Put" method="PUT"
Service fn="CM_Move" method="MOVE"
Service fn="CM_MkDir" method="MKDIR"
```

Figure 8. Partial `obj.conf` **Configuration File for NES 3.6**

5

5. ServletExec/AS:
Application Server (Windows)

Installing the Windows 2000/Windows NT web server adapters for Microsoft IIS, Netscape Enterprise Server, iPlanet Web Server, and Apache HTTP Server

T he ServletExec Application Server (ServletExec/AS) is a standalone web application server that hosts Java Servlets and JavaServer Pages (JSP). ServletExec/AS does not contain a built-in HTTP (web) server. Therefore, it must be used in conjunction with Microsoft IIS, Netscape Enterprise Server (NES), iPlanet Web Server (iWS), or Apache HTTP Server. Web server adapters for these web servers are included in the ServletExec/AS installer.

The ServletExec/AS for Windows installer is named `ServletExec_AS_41.exe`. If you haven't already done so, you can download the installer file from:

http://www.newatlanta.com/downloads/index.jsp

This chapter contains important information that will allow you to verify your installation of ServletExec/AS for Windows. It will also be useful should you decide to uninstall ServletExec/AS and need to make sure you've completely removed all installed components.

5.1 Upgrading from a Previous Version

You can upgrade to ServletExec 4.1 from all previous versions except versions 3.0C or 3.0E. If you have either of these versions installed, you will need to completely uninstall that version and do a clean installation of ServletExec 4.1.

ServletExec/AS 3.0 and later replaces the ServletExec/Apache and ServletExec/WAI implementations of ServletExec 2.2 and earlier. However, the ServletExec/AS configuration files are slightly different from those for ServletExec/Apache and ServletExec/WAI. Therefore, it is not possible to automatically upgrade from ServletExec/Apache or ServletExec/WAI to ServletExec/AS. Instead, you must do a clean install of ServletExec/AS and manually re-enter your configuration data using the ServletExec Admin UI.

To upgrade to ServletExec/AS 4.1 from 3.0 or 3.1 and maintain your old configuration settings

1. Stop your web server. If you've installed ServletExec/AS adapters for multiple web servers, stop each one that is running.
2. Make backup copies of the `ServletExec Data`, `Servlets`, `classes` (if you've added an additional one), and `webapps` (versions 3.1 and above only) subdirectories for all ServletExec/AS instances.

   - For ServletExec/AS 3.0 and higher, the default locations are within:

     `C:\Program Files\New Atlanta\ServletExec AS\se-<instance-name>`

3. Close all open applications, making sure to include DBMON.
4. Uninstall each of the old versions of ServletExec/AS. In Control Panel, click **Add/Remove Programs** and follow the instructions on the **Install/Uninstall** tab, choosing To uninstall ServletExec/AS or a web server adapter when prompted.
5. Run the ServletExec/AS 4.1 installer. You will need to re-run the installer for each ServletExec/AS instance.

   > **Note**
   >
   > The default installation directory for versions 2.2 and earlier is different.

   > **Important**
   >
   > For a successful upgrade, you must install everything into the same directory as the previous version.

6. Copy the `ServletExec Data`, `Servlets`, `classes` (if you've added an additional one), and `webapps` (versions 3.1 and above only) subdirectories to the new `ServletExec AS` directory created by the installer in Step 5. In ServletExec/AS 4.1, the default location of the `ServletExec AS` directory is:

   `C:\Program Files\New Atlanta\ServletExec AS`

7. Restart your web server.

After restarting your web server, the new version of ServletExec/AS will run using your old configuration settings.

> **Tip**
>
> If you have any problems, you can restore the subdirectories from the backups you made in Step 3.

5.2 System Requirements

5.2.1 Operating System and Web Server

- ServletExec/AS 4.1 is not supported on Windows 95, Windows 98 or Windows ME.

- ServletExec/AS 4.1 supports the following operating systems and web servers through the use of web adapters:

OS	Web Server
Windows® 2000	▪ Microsoft IIS 5.0 ▪ Netscape Enterprise Server (NES) 3.5.1 and higher ▪ iPlanet Web Server (iWS) 4.0 and higher ▪ Apache HTTP Server 1.3.9 through 1.3.20
Windows NT® 4.0 *Service Pack 4 or higher*	▪ Microsoft IIS 4.0 ▪ Netscape Enterprise Server (NES) 3.5.1 and higher ▪ iPlanet Web Server (iWS) 4.0 and higher ▪ Apache HTTP Server 1.3.9 through 1.3.20

Table 5. AS-Windows Supported Operating Systems and Web Servers

5.2.2 JDK or JRE

ServletExec/AS 4.1 is not supported on JDK or JRE 1.1.x or earlier.

To install ServletExec/AS 4.1, you must first install one of these Java Development Kit (JDK™) or the Java™ Runtime Environment (JRE) choices:

- JDK/JRE 1.2.2 or 1.3.1 for Windows from Sun Microsystems
- JDK/JRE for Windows 1.3 from IBM

  Important

  The JRE does not contain a Java compiler, which is required by ServletExec to support JavaServer Pages (JSP). Be sure to install the JDK, and not the JRE, if you plan to use JSP.

You can download the Sun JDKs for Windows from:

http://java.sun.com/j2se/

You can download the IBM JDKs for Windows from:

http://www.ibm.com/developerworks/java/

As of this writing, the latest production versions of the Sun JDKs for Windows are 1.2.2 and 1.3.1. The latest production version of the IBM JDK for Windows is 1.3. These are the versions we tested and recommend using with ServletExec/NSAPI 4.1.

5.2.3 JDK/JRE Installation

To install ServletExec/AS, you must have first installed a compatible version of the JDK or JRE as explained in Section 5.2.2.

Important

The JDK or JRE must be installed on a local drive and not on a mapped network drive. If the JDK/JRE has been installed on a mapped network drive, ServletExec/AS will not be able to load and initialize the Java VM.

5.2.3.1 Multiple JDK/JRE Versions Installed

You may have multiple versions of the JDK/JRE installed on your system. The ServletExec/AS installer uses registry entries to determine which installed JDK/JRE to use. It will look for an installed JDK first and if it doesn't find one, it will look for a JRE. It will look for JDK/JREs from vendors in the following order: Sun Microsystems, and then IBM. Study the complete algorithm below, to gain a better understanding.

How ServletExec selects an installed JDK/JRE

1. Look for a Sun JDK using the following registry key:

   `HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Development Kit`

   - If Sun JDK found, go to Step 2. Otherwise, look for a Sun JRE using the following key:

     `HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Runtime Environment`

   - If Sun JRE found, go to Step 2. Otherwise, look for an IBM JDK or JRE using the following keys .

     `HKEY_LOCAL_MACHINE\SOFTWARE\IBM\Java Development Kit`

     or…

     `HKEY_LOCAL_MACHINE\SOFTWARE\IBM\Java Runtime Environment`

   - If no Sun or IBM JDK/JRE found using the above registry keys, then abort ServletExec installer.

2. Read the `CurrentVersion` variable from the key found in Step 1. Currently, the only valid values for this variable start with "1.2" or "1.3" (including, for example "1.2.2").

3. Append the value of the `CurrentVersion` variable from Step 2 to the key from Step 1 to create a new key. For example:

   `HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Development Kit\1.2`

   or…

   `HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Runtime Environment\1.2`

4. Read the value of the `JavaHome` variable for the key from Step 3 to find the location of the JDK or JRE.

5.3 Uninstalling Other Servlet Engines

Very Important

You **must** uninstall any other servlet engines previously installed before installing and using ServletExec/AS. In particular, modifications to the `obj.conf` and `httpd.conf` configuration files for other servlet engines must be removed.

▪ For Microsoft IIS, the ISAPI Filters metabase entries associated with other servlet engines must be removed.
See Section 5.5.3.2 for more information on ISAPI filters metabase entries.

▪ For Netscape Enterprise Server and iPlanet Web Server, make sure the internal servlet and JSP support is disabled.
See Section 3.3 for information on disabling servlet engines.

▪ For Apache HTTP Server, any directives associated with other servlet engines in `httpd.conf` must be removed.
See Section 5.5.5.2 for more information on `httpd.conf` entries.

5.4 Running the ServletExec/AS Installer

To run the ServletExec/AS Installer

1. Double-click the ServletExec/AS installer icon. After unpacking, the installer will display the ServletExec 4.1 Application Server Setup window.
2. Click **Next** to bring up the Setup Type window.

   After clicking **Next** on the welcome window, you will be prompted to choose a setup type, as shown in Figure 9.

   The first time you run the ServletExec/AS installer, you will be offered two options for setup type. For subsequent installations, you will have additional options.

   Typically, you will choose **Install a ServletExec/AS instance**, which will install a single ServletExec/AS instance and a single web server adapter.

   Important

   You must install at least one web server adapter whenever you install a ServletExec/AS instance.

   The only time you would choose **Install a web server adapter** is after you have previously installed a ServletExec/AS instance on a different physical machine from the web server, and now wish to install a ServletExec/AS adapter to allow the web server to communicate with the ServletExec/AS instance running on the remote machine.

   The remainder of this section applies only if you have selected **Install a ServletExec/AS instance**.
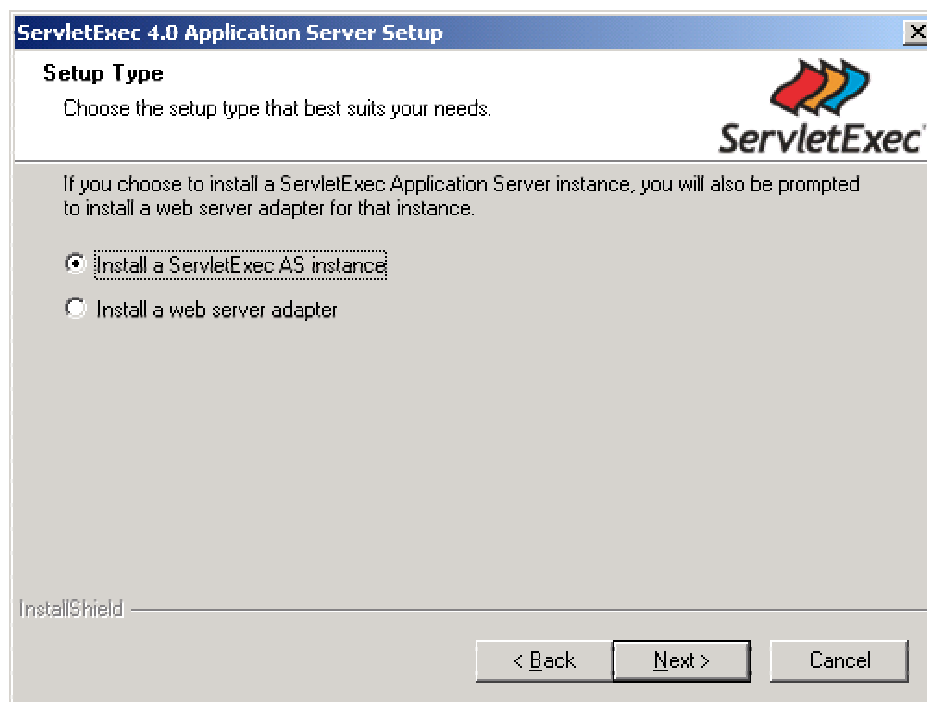
Figure 9. Setup Type

3. Click **Next**. Review the license agreement carefully and check the READ ME for any last minute notices before continuing.
4. After accepting the license agreement and reviewing the READ ME, the ServletExec/AS installer prompts you for the destination folder for installing the ServletExec/AS files. You may choose the default or click **Browse** to select a different folder.

Important

Do NOT move or rename the folder you choose after ServletExec/AS is installed.

---

5. Enter an **Instance Name**. After selecting a destination folder, the ServletExec/AS installer prompts you for the ServletExec/AS instance name. This name is used to uniquely identify the ServletExec/AS instance for administration, and is especially important if you install more than one instance on your server. By default, the installer suggests using the computer name as defined in Control Panel's **Network**, but you're free to use any unique name. Avoid using spaces or other special characters (">", "/", etc.) in the instance name.
6. Select a web server. After defining the ServletExec/AS instance name, the installer prompts you to select a web server on which to install a web server adapter for communicating with the ServletExec/AS instance that is about to be installed. The selection dialog is illustrated in Figure 10.

If you select Netscape Enterprise Server (NES) or iPlanet Web Server (iWS), the ServletExec/AS installer will then prompt you to select a server instance from a

list of configured servers (both NES and iWS support multiple servers per physical machine).
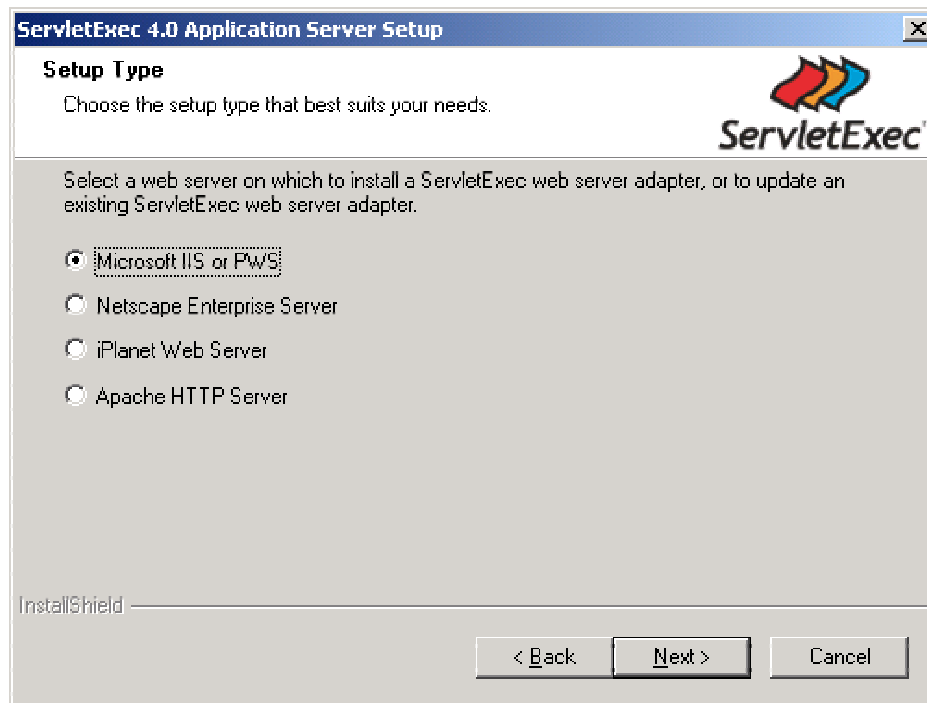
Figure 10. Web Server Adapter

7. Before making any change to your computer, the ServletExec/AS installer displays a confirmation dialog, as represented in Figure 11. Click **Back** to modify any of your selections, or **Cancel** to abort the installation. After clicking **Next**, the ServletExec/AS installer will begin copying files to your computer.

8. Install ServletExec as an NT Service? The installer will prompt you to install the ServletExec/AS instance as an NT service. Installing the instance as an NT service has the following advantages:

- You will be able to start and stop the ServletExec/AS instance from the **Services** control panel.

- The ServletExec/AS instance will continue running if you log off Windows.

- The ServletExec/AS instance can be configured using the **Services** control panel to automatically start when Windows restarts.

If you choose not to install the ServletExec/AS instance as an NT Service, you will need to manually start and stop the instance as described in Section 5.6, below.
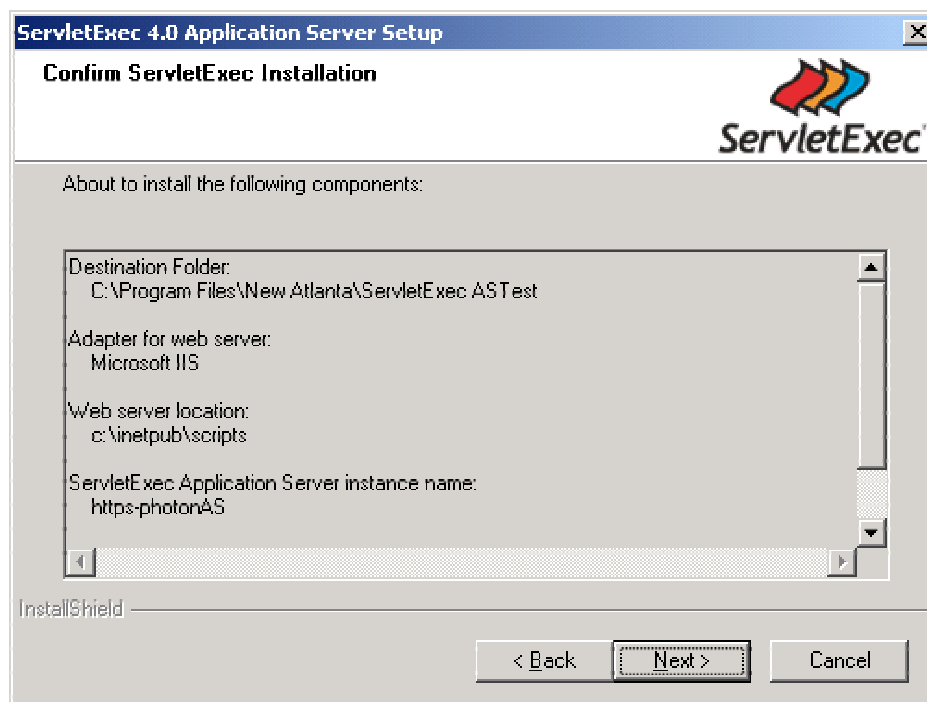
Figure 11. Confirm ServletExec Installation

9. Update the server configuration file. If you're installing a web server adapter for Netscape Enterprise Server, iPlanet Web Server, or Apache HTTP Server, the ServletExec/AS installer will prompt you to update the server configuration file. The web server configuration file **must** be updated for the ServletExec/AS web server adapter to work properly, so it is recommended that you allow the installer to do so. If you choose not to allow the installer to update the web server configuration file you must do so manually as described in Section 5.5.

If you're installing a web server adapter for Microsoft IIS, the ServletExec/AS installer will automatically create the servletexec.properties configuration file. After completing the ServletExec/AS installation, be sure to restart the web server on which you installed the ServletExec/AS web server adapter.

5.5 What Was Installed and/or Modified?

When you installed ServletExec/AS, these changes were made to your system:

- The ServletExec AS directory was created

- ServletExec/AS registry entries were created or modified

- The web server configuration file was modified to support the ServletExec/AS web server adapter

---

- The web server configuration file was modified to enable Connection Pooling between the web server adapter and ServletExec

The following sections describe each of these changes.

5.5.1 The ServletExec AS Directory

The `ServletExec AS` directory was created in the location you selected during the installation process. The default location suggested by the installer is:

```
C:\Program Files\New Atlanta\ServletExec AS
```

Important

Do not move the `ServletExec AS` directory after installation. If you move this directory, ServletExec will be unable to find its configuration files.

There are no restrictions on the location of the `ServletExec AS` directory.

The `ServletExec AS` directory contains the following subdirectories:

`bin`	The `bin` subdirectory contains the native code web server adapters used by ServletExec/AS. It also contains utility programs for installing and uninstalling a ServletExec/AS instance as a Windows service.
	Important
	Do not modify the contents of this directory!
`Documentation`	Contains the ServletExec User Guide, and the Servlet 2.3 and JSP 1.2 API documentation from Sun Microsystems.
`Examples`	This subdirectory contains examples of a web application and a JSP page with an embedded bean.
	See the appropriate chapters of the ServletExec User Guide for instructions on running these examples.
`ExampleServlets`	The `ExampleServlets` subdirectory contains example servlets that are copied to the `Servlets` subdirectory for each ServletExec/AS instance as it is installed. The example servlets are here just for use by the installer.
`se-<instance-name>`	There is one `se-<instance-name>` subdirectory for each ServletExec/AS installed on the computer.
	See further discussion of this directory directly below.
`installerFiles`	The `installerFiles` subdirectory contains files used by the ServletExec/AS setup program for installing multiple ServletExec/AS instances. This directory and

its contents are only used by the installer.

`lib`	The `lib` subdirectory contains the Java archive (.jar) files required by ServletExec/AS.

Important

Do not modify the contents of this directory!

The `se-<instance-name>` subdirectories of the `ServletExec AS` directory each contain the following subdirectories.

`classes`	The `classes` subdirectory is automatically added to the ServletExec VM classpath. Place class files that will be shared by multiple servlets in this directory. *See the READ ME in the `classes` folder for more information.*
`Servlet Logs`	This subdirectory contains the `Servlet.log` files that capture the output from the servlet `log()` method. In some cases, this directory isn't created until the first ServletExec initialization.
`ServletExecData`	Contains the ServletExec/AS configuration files. These files are generally modified using the ServletExec Admin UI, but can also be edited manually.
`Servlets`	This subdirectory contains the class files for servlets hosted by ServletExec/AS. *See the READ ME in the `Servlets` directory and the ServletExec User Guide for more information.*
`webapps`	One of these subdirectories exists for each virtual server configured. This subdirectory contains the "default" folder for automatically deploying web applications hosted by ServletExec/AS. *See the READ ME in the `webapps` directory and the ServletExec User Guide for more information.*

5.5.2 Registry Entries

The ServletExec/AS registry entries are stored under the following key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\New Atlanta Communications\ServletExec AS
```

Important

These registry entries are modified by the ServletExec/AS installer and uninstaller, and should not be edited manually.

5.5.3 Microsoft IIS

If you installed a web server adapter for Microsoft IIS, the installer modified your system as described in the following sections.

5.5.3.1 ServletExec_Adapter.dll

The `ServletExec_Adapter.dll` dynamic link library (DLL) was installed in the `C:\InetPub\Scripts` directory, which is mapped to the `SCRIPTS` virtual directory in the default IIS installation; or, `ServletExec_Adapter.dll` was copied to the directory you selected during the installation process if the `SCRIPTS` virtual directory is not defined. The directory in which `ServletExec_Adapter.dll` resides **must** be mapped to a Microsoft IIS virtual directory and the virtual directory **must** have `execute` permission enabled. Configure virtual directories for Microsoft IIS a using the Microsoft Management Console (Internet Service Manager).

5.5.3.2 Metabase ISAPI Filter Entry

The ServletExec/ISAPI installer automatically modifies the IIS 4.0 (Windows NT 4.0) or IIS 5.0 (Windows 2000) metabase to add the ISAPI Filter entry.

Very Important

You **must** remove entries for other servlet engines you may have previously installed. The uninstallers for most servlet engines do not automatically remove the metabase ISAPI Filter entry.

To examine or manually modify the metabase ISAPI Filter under IIS 5.0

1. Click **Start**, point to **Settings** and click **Control Panel**.
2. In Control Panel, double-click **Administrative Tools**.
3. In Administrative Tools, double-click **Internet Services Manager**.
4. Expand the **Internet Services Manager** entry until you can see the icon for your server.
5. Right-click the server icon, and then click **Properties**. You should see a dialog similar to Figure 14. Make sure you've opened the **Properties** dialog for the server as illustrated in Figure 14, and not for the **Default Web Site** (or any other web site you may have configured).
6. In **Properties**, check to make sure **WWW Service** is selected in the **Master Propertie**s list (*see* Figure 14), and then click **Edit**.
7. In **Master Properties**, click the **ISAPI Filters** tab (*see Figure 15*).
8. Click any previously installed servlet engines, and then click **Remove**.
9. Click **ServletExec,** and then click **Edit** to examine or modify it (if you wish). If you do not see **ServletExec**, click **Add** and follow the instructions to add the ServletExec/ISAPI entry, making sure to point to `ServletExec_AS.dll`.
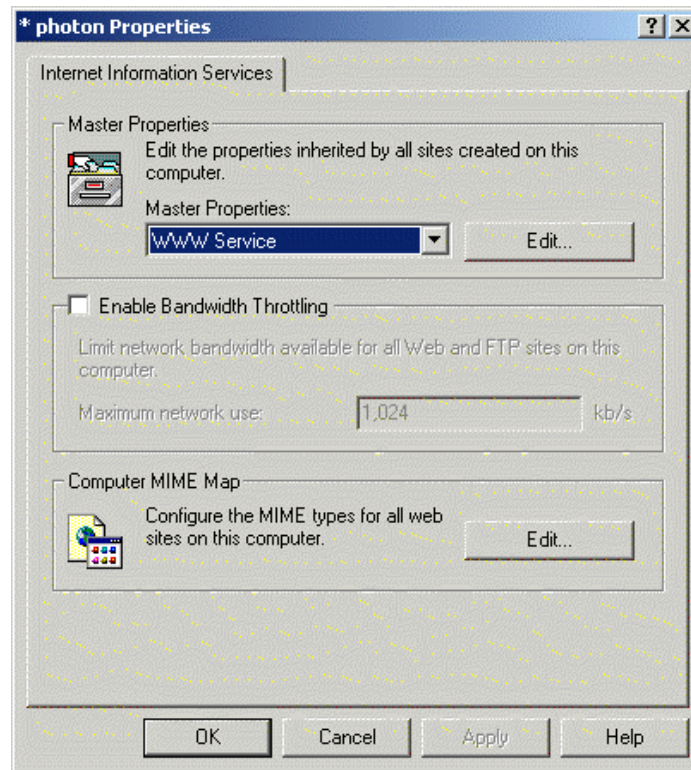
Figure 12. Server Properties (IIS 5.0)
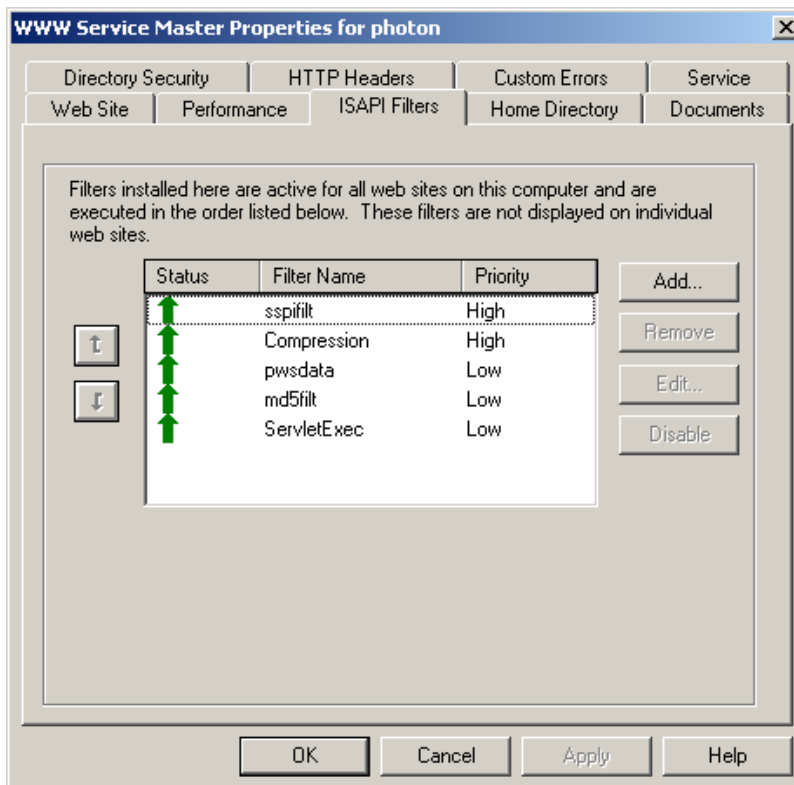
Figure 13. ISAPI Filters (IIS 5.0)

To examine or manually modify the metabase ISAPI Filter under IIS 4.0

1. Open Internet Service Manager (Microsoft Management Console) and expand the **Internet Information Server** entry until you can see the icon for your server.

2. Right-click the server icon, and then click **Properties**. You should see a dialog similar to Figure 14. Make sure you've opened the **Properties** dialog for the server as illustrated in Figure 14, and not for the **Default Web Site** (or any other web site you may have configured).

3. In **Properties**, check to make sure **WWW Service** is selected in the **Master Propertie**s list (*see* Figure 14), and then click **Edit**.
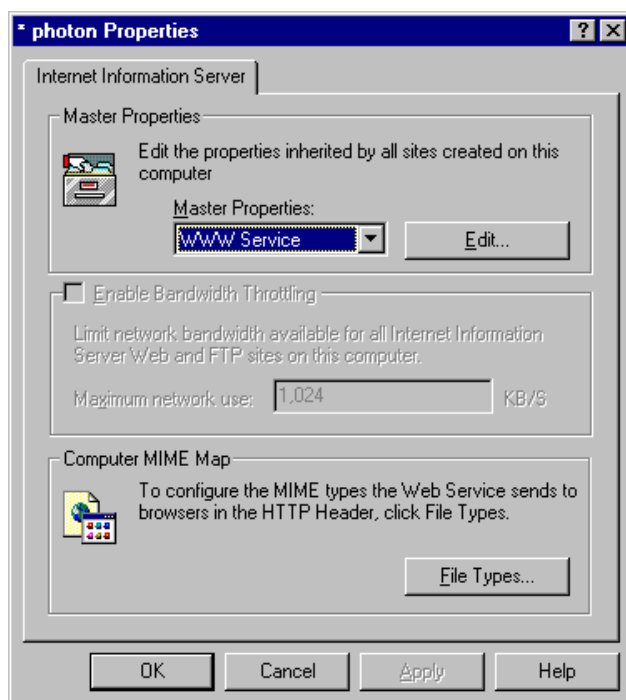
Figure 14. Server Properties (IIS 4.0)

4. In **Master Properties**, click the **ISAPI Filters** tab (*see Figure 15*).

5. Click any previously installed servlet engines, and then click **Remove**.

6. Click **ServletExec,** and then click **Edit** to examine or modify it (if you wish). If you do not see **ServletExec**, click **Add** and follow the instructions to add the ServletExec/ISAPI entry, making sure to point to `ServletExec_AS.dll`.
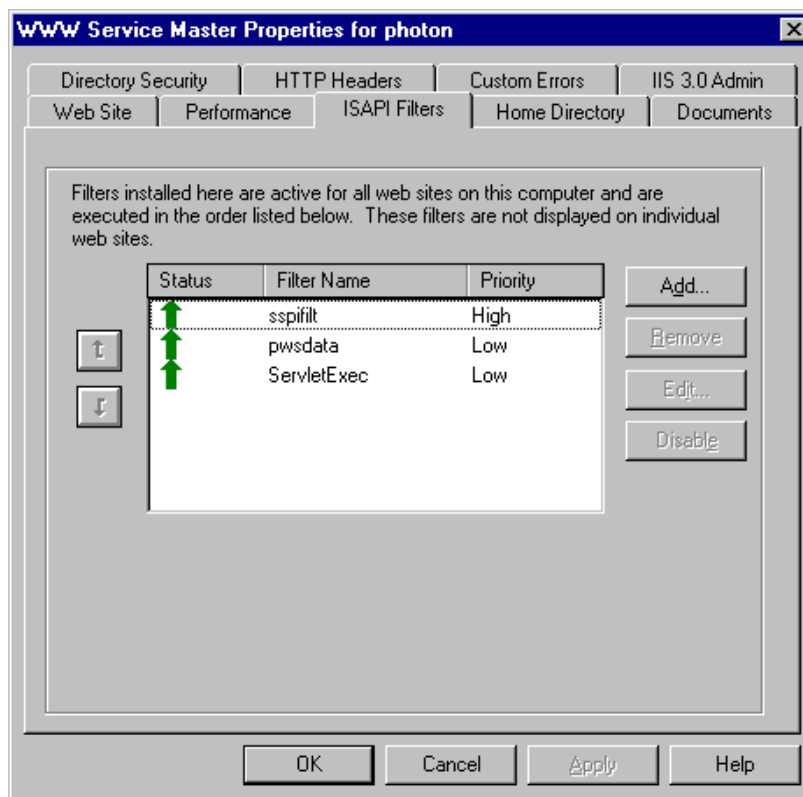
Figure 15. ISAPI Filters

5.5.3.3 The `servletexec.properties` Configuration File

The `servletexec.properties` file was created in the `C:\InetPub\scripts` directory (or, in the same directory as `ServletExec_Adapter.dll`, if the DLL was installed in some other directory). This file contains the configuration properties used by the ServletExec adapter for the Microsoft IIS/PWS web server. These properties are used by the adapter for routing HTTP requests from the web server to a ServletExec Application Server (AS) instance.

Connection Pooling Between the Web Server Adapter and ServletExec

The installer automatically enables connection pooling between the web server adapter and ServletExec, using default settings. The purpose of this connection pooling is to improve performance through better resource management.

You can change the default settings by modifying the configuration file as explained below. The two parameters that provide the connection pooling are: `pool max idle` (sets maximum idle time for a connection… default setting is 10) and `pool increment.` (sets connection increment value… default setting is 5).

A good guideline for changing these parameters from their default settings would be to set the `pool max idle` to your average number of connections, and to set the `pool increment` to the average spike number of connections.

To modify Connection Pooling in IIS for ServletExec/AS

- Add the following to the end of the `servletexec.properties` file on two separate lines:

  ```
  servletexec.<instance-name>.pool-increment=<value>
  servletexec.<instance-name>.pool-max-idle=<value>
  ```

5.5.3.4 Web Application URL Context

For each web application URL context configured using the ServletExec Admin UI, you must update the *applications* property in the `servletexec.properties` file.

To update the applications property in `servletexec.properties`

- Use the following form:

  ```
  servletexec.<instance-name>.applications=/URL-context1, /URL-context2
  ```

5.5.3.5 Prefix Aliases

For each prefix alias configured using the ServletExec Admin UI, you must update the *aliases* property in the `servletexec.properties` file.

To update the aliases property in `servletexec.properties`

- Use the following form:

  ```
  servletexec.<instance-name>.aliases=/prefix-alias1, /prefix-alias2
  ```

5.5.3.6 Suffix Aliases

For each suffix alias configured using the ServletExec Admin UI, you must update the *aliases* property in the `servletexec.properties` file.

To update the aliases property in `servletexec.properties`

- Use the following form:

  ```
  servletexec.<instance-name>.aliases=/suffix-alias1, /suffix-alias2
  ```

5.5.4 Netscape Enterprise Server / iPlanet Web Server

If you installed a web server adapter for Netscape Enterprise Server (NES) or iPlanet Web Server (iWS), the installer modified your system as described in the following sections.

5.5.4.1 Server Configuration File (`obj.conf`)

To install a ServletExec/AS web server adapter for NES/iWS, modifications must be made to the web server's `obj.conf` file. This section describes the modifications to `obj.conf` made by the ServletExec/AS installer.

Important

If you chose not to allow the installer to make these modifications, you must make them manually.

Several lines must be added to `obj.conf` for each server for which ServletExec/AS is installed (the location of these lines within the `obj.conf` file is very important):

1. Add the following lines to the beginning of `obj.conf` before the other `Init` directives:

```
Init    fn="load-modules"
        shlib="<path>/bin/nsapi/ServletExec_Adapter.dll"
        funcs="ServletExecInit,ServletExecService"
Init    fn="ServletExecInit" <instance-name>.instances="127.0.0.1:8888"
```

where <path> is the full path to the ServletExec/AS installation directory; by default, this is `C:\Program Files\New Atlanta\ServletExec AS`.

Note:

The first `Init` directive will normally appear on a single line within the `obj.conf` file. It's shown as spanning three lines here for formatting reasons. It may span three lines within `obj.conf`, in which case the second and third lines must begin with a tab or space character). In general, the format of the second `Init` directive is:

```
Init fn=ServletExecInit <instance-name>.instances="<addr>:<port>"
```

where `<instance-name>`, `<addr>`, and `<port>` are the name, IP address, and port number of the ServletExec/AS instance to which HTTP requests will be forwarded.

Important—iWS 6.0 or Higher Users

If you're using iWS 6.0 or higher, you must add the `Init` directives to the `magnus.conf` file rather than the `obj.conf` file.

---

2. The following directives must be added after the <Object name=default> directive:

```
<Object name="<instance-name>">
Service fn="ServletExecService" group="<instance-name>"
</Object>
```

In general, an `Object` directive must be added to `obj.conf` for every ServletExec/AS instance to which the web server adapter will forward requests. The `Object` directives have the form...

```
<Object name="<instance-name>">
Service fn="ServletExecService" group="<instance-name>"
</Object>
```

where `<instance-name>` is the name of the ServletExec/AS instance.

3. Lines similar to the following must be added within the <Object name=default> directives:

```
NameTrans fn="assign-name" from="/servlet/*" name="<instance-name>"
NameTrans fn="assign-name" from="*.jsp*" name="<instance-name>"
```

These `NameTrans` directives define URL prefix and suffix mappings that cause HTTP requests to be forwarded to a ServletExec/AS instance for processing.

Figure 16 shows a complete `obj.conf` file for NES 3.6 with the ServletExec/AS directives highlighted.

Important

You must stop and restart the NES/iWS web server after modifying the `obj.conf` configuration file.

```
Init fn="load-modules"
    shlib="C:/Program Files/New Atlanta/ServletExec AS/bin/nsapi/ServletExec_Adapter.dll"
    funcs="ServletExecInit,ServletExecService"

Init fn="ServletExecInit" <instance-name>.instances="127.0.0.1:8888"

Init fn=flex-init
    access="D:/Netscape/SuiteSpot/https-<instance-name>/logs/access"
    format.access="%Ses->client.ip% - %Req->vars.auth-user% [%SYSDATE%] \"%Req->reqpb.clf-request%\" %Req->srvhdrs.clf-status% %Req->srvhdrs.content-length%"

Init fn=load-types mime-types=mime.types

<Object name=default>
NameTrans fn="assign-name" from="/example/*" name="<instance-name>"
NameTrans fn="assign-name" from="/servlet/*" name="<instance-name>"
NameTrans fn="assign-name" from="*.jsp*" name="<instance-name>"
NameTrans fn=pfx2dir from=/ns-icons dir="D:/Netscape/SuiteSpot/ns-icons"
NameTrans fn=pfx2dir from=/mc-icons dir="D:/Netscape/SuiteSpot/ns-icons"
NameTrans fn="pfx2dir" from="/help" dir="D:/Netscape/SuiteSpot/manual/https/ug"
NameTrans fn=document-root root="D:/Netscape/SuiteSpot/docs"
PathCheck fn=nt-uri-clean
PathCheck fn="check-acl" acl="default"
PathCheck fn=find-pathinfo
PathCheck fn=find-index index-names="index.html,home.html"
ObjectType fn=type-by-extension
ObjectType fn=force-type type=text/plain
Service method=(GET|HEAD) type=magnus-internal/imagemap fn=imagemap
Service method=(GET|HEAD) type=magnus-internal/directory fn=index-common
Service method=(GET|HEAD) type=*~magnus-internal/* fn=send-file
AddLog fn=flex-log name="access"
</Object>

<Object name=cgi>
ObjectType fn=force-type type=magnus-internal/cgi
Service fn=send-cgi
</Object>

<Object name="<instance-name>">
Service fn="ServletExecService" group="<instance-name>"
</Object>
```

Figure 16. Complete `obj.conf` Configuration File for NES 3.6

Connection Pooling Between the Web Server Adapter and ServletExec

The installer automatically enables connection pooling between the web server adapter and ServletExec, using default settings. The purpose of this connection pooling is to improve performance through better resource management.

You can change the default settings by modifying the configuration file as explained below. The two parameters that provide the connection pooling are: `pool max idle` (sets maximum idle time for a connection… default setting is 10) and `pool increment`. (sets connection increment value… default setting is 5).

A good guideline for changing these parameters from their default settings would be to set the `pool max idle` to your average number of connections, and to set the `pool increment` to the average spike number of connections.

To modify Connection Pooling in NES/iWS for ServletExec/AS

- Add the following to the beginning of the `obj.conf` file. Go back to Step 1 earlier in this section, copy it that line and insert a space at the end it, adding the following immediately after and on the same line (the formatting requirements of this page are wrapping the text to fit the page):

   ```
   <instance-name>.pool-max-idle=<value> <instance-name>.pool-
   increment=<value>
   ```

5.5.4.2 Web Application URL Context

For each web application URL context configured using the ServletExec Admin UI, you must add a *NameTrans* directive to the `obj.conf` file.

To update the applications property in `obj.conf`

Use the following form:

```
NameTrans fn="assign-name" from="/URL-context/*" name="<instance-name>"
```

5.5.4.3 Prefix Aliases

For each prefix alias configured using the ServletExec Admin UI, you must add a *NameTrans* directive to in the `obj.conf` file.

To update the aliases property in `obj.conf`

Use the following form:

```
NameTrans fn="assign-name" from="/prefix-alias/*" name="<instance-name>"
```

5.5.4.4 Suffix Aliases

For each suffix alias configured using the ServletExec Admin UI, you must add a *NameTrans* directive to the `obj.conf` file.

To update the aliases property in `obj.conf`

Use the following form:

```
NameTrans fn="assign-name" from="/suffix-alias/*" name="<instance-name>"
```

5.5.5 Apache HTTP Server

If you installed a web server adapter for Apache HTTP Server, the installer modified your system as described in the following sections.

5.5.5.1 ApacheModuleServletExec.dll

The ServletExec/AS web server adapter for Apache HTTP Server is implemented by the `ApacheModuleServletExec.dll` library, which is copied by the ServletExec/AS installer to the `modules` subdirectory of the Apache installation directory.

5.5.5.2 Server Configuration File (httpd.conf)

To install a ServletExec/AS web server adapter for Apache, modifications must be made to the web server's `httpd.conf` file. This section describes the modifications to `httpd.conf` made by the ServletExec/AS installer.

Important

If you chose not to allow the installer to make these modifications, you must make them manually.

The following directive was added to `httpd.conf` with the other `LoadModule` directives (near the top of the file):

```
LoadModule servletexec_module modules/ApacheModuleServletExec.dll
```

This directive causes Apache to load the ServletExec/AS web server adapter.

The following lines were added to the end of the `httpd.conf` file:

```
ServletExecInstances <instance-name> 127.0.0.1:8888
ServletExecAliases <instance-name> /servlet .tmpl .jsp
<Location /servlet>
    SetHandler servlet-exec
</Location>
AddHandler servlet-exec jsp
```

These directives define the default prefix and suffix aliases used by ServletExec/AS

See the ServletExec User Guide for a general discussion of servlet aliases, as well as further discussion of configuring prefix and suffix aliases with Apache, in Section 5.5.5.3 and Section 5.5.5.5.

Connection Pooling Between the Web Server Adapter and ServletExec

The installer automatically enables connection pooling between the web server adapter and ServletExec, using default settings. The purpose of this connection pooling is to improve performance through better resource management.

You can change the default settings by modifying the configuration file as explained below. The two parameters that provide the connection pooling are: `pool max idle` (sets maximum idle time for a connection… default setting is 10) and `pool increment`. (sets connection increment value… default setting is 5).

A good guideline for changing these parameters from their default settings would be to set the `pool max idle` to your average number of connections, and to set the `pool increment` to the average spike number of connections.

To modify Connection Pooling in Apache HTTP Server for ServletExec/AS (Windows)

- Add the following to the end of the `http.conf` file on two separate lines:
    ```
    ServletExecPoolIncrement <instance-name> <value>
    ServletExecPoolMaxIdle <instance-name> <value>
    ```

5.5.5.3 Web Application URL Context

For each web application URL context configured using the ServletExec Admin UI, you must update the *ServletExecApplications* directive in the `httpd.conf` file.

To update the ServletExecApplications directive in `httpd.conf`

- Use the following form:
  ```
  ServletExecApplications <instance-name> /URL-context1 /URL-context2
  <Location /URL-context1>
                  SetHandler servlet-exec
  </Location>
  <Location /URL-context2>
                  SetHandler servlet-exec
  </Location>
  ```

5.5.5.4 Prefix Aliases

For each prefix alias that is configured using the ServletExec Admin UI, the `ServletExecAliases` directive must be updated, and a `Location` directive must be added to `httpd.conf` in the following format:

```
ServletExecAliases <instance-name> /prefix-alias
<Location /prefix-alias>
    SetHandler servlet-exec
</Location>
```

If you allowed the ServletExec/AS installer to automatically update `httpd.conf`, ServletExec/AS is configured to handle the `/servlet` prefix alias by default.

5.5.5.5 Suffix Aliases

There are two methods for configuring suffix aliases with Apache. The most common method is illustrated below. For each suffix alias that is configured using the ServletExec Admin UI, the `ServletExecAliases` directive must be updated, and an `AddHandler` directive must be added to `httpd.conf` with the following format:

```
ServletExecAliases <instance-name> <suffix-aliases>
AddHandler servlet-exec <suffix-alias>
```

You should use this method if the URL Rewriting option of the Session Tracking feature is disabled, which it is by default. This method of configuring suffix aliases provides better performance than the alternative method.

See the ServletExec User Guide for more information on Session Tracking and URL Rewriting.

URL Rewriting Alternative

If you enabled the URL Rewriting option of the Session Tracking feature, you must use an alternative method of configuring suffix aliases. For this method, you use a Location directive instead of the AddHandler directive. For each suffix alias that is configured using the ServletExec Admin UI, add a `Location` directive to `httpd.conf` with the following format:

```
<Location /*.suffix-alias*>
SetHandler servlet-exec
</Location>
```

In addition, for each subdirectory level beneath the web server document root directory, you must add an additional Location directive to `httpd.conf`. For example, assume the

---

web server root document directory is `C:\Program Files\Apache Group\Apache\htdocs` and the following subdirectories appear beneath `htdocs`:

```
htdocs\subdir1
htdocs\subdir2
htdocs\subdir2\subdir3
```

In this example, there are two levels of subdirectories beneath `htdocs`, so to create a suffix alias of `.jsp`, you should add three Location directives to `httpd.conf`, one for `htdocs` and one for each subdirectory level:

```
#
# suffix alias for htdocs
#
<Location /*.jsp*>
SetHandler servlet-exec
</Location>
#
# suffix alias for htdocs\subdir1 and htdocs\subdir2
#
<Location /*/*.jsp*>
SetHandler servlet-exec
</Location>
# (continued on next page)
#
# suffix alias for htdocs\subdir2\subdir3
#
<Location /*/*/*.jsp*>
SetHandler servlet-exec
</Location>
```

5.6 ServletExec/AS Operation

This section provides instructions for starting and stopping ServletExec/AS, for configuring a ServletExec/AS instance to run on a remote machine from the web server, and for configuring multiple ServletExec/AS instances for a web server.

If you installed a ServletExec/AS instance as a Windows service, you can start and stop the instance using the **Services** control panel. **Services** relies on the `StartServletExec.bat` and `StopServletExec.bat` files, which are located in the `se-<instance-name>` subdirectories of the `ServletExec AS` installation directory, to start and stop the instance. Do not move or delete these batch files.

Important

If you set or modify the ServletExec Admin UI username and password, you **must** manually update the `StopServletExec.bat` command file.

See Section 5.6.2 for further instructions.

5.6.1 Starting ServletExec/AS

A ServletExec/AS instance is a standalone Java application that communicates with the web server (through the ServletExec/AS web server adapter) using network sockets. You start a ServletExec/AS instance using the `java` command on the command line. In

---

addition to `java` command arguments, ServletExec/AS accepts several required and optional arguments. You can also start a ServletExec/AS instance using the `StartServletExec.bat` file. The following sections describe both methods.

5.6.1.1 StartServletExec.bat

The batch file `StartServletExec.bat` is provided for your convenience in starting the ServletExec/AS instance. It is also used by **Services** to start the ServletExec/AS instance.

`StartServletExec.bat` includes defaults for all `java` and ServletExec/AS command line arguments. You can edit `StartServletExec.bat` to modify them. The next section features a complete list of ServletExec/AS command line arguments.

5.6.1.2 Using the java Command

We recommend you use the `StartServletExec.bat` file or **Services** to start a ServletExec/AS instance. However, it is possible to start a ServletExec/AS instance from a DOS command prompt using the `java` command. The information in this section can also help you edit the `StartServletExec.bat` file to modify the startup parameters for a ServletExec/AS instance.

To start a ServletExec/AS instance using the `java` command

- From the command line, type the following:

    ```
    java [java arguments] ServletExec [ServletExec/AS arguments]
    ```

Important

The `java` command arguments must include the `-classpath` option specifying the path to the `ServletExec41.jar`, `servlet.jar`, `crimson.jar`, `jaxp.jar`, and `jndi.jar` archives.

Start ServletExec/AS arguments

If an option value contains spaces then the entire value must be placed inside double quotes, as in the following example:

    ```
    -name "Local Instance"
    ```

See the following table for a complete list of available arguments.

Argument	Explanation
`-help`	Displays all available arguments and explanations.
`-port <port number>`	**Optional**. The default value is 8888. This argument specifies the TCP/IP port on which the ServletExec/AS instance communicates with the web server adapter(s). If you modify this value, you must also modify the web server configuration file(s) as described above.
`-backlog <length>`	**Optional**. The default value is 50. This

Argument	Explanation
	argument specifies the size of the ServletExec/AS incoming request queue. Requests received when the queue is full are rejected. Increasing the size of the request queue may prevent requests from being rejected under heavy loads, but may increase per-request response times.
`-name <instance-name>`	**Optional**. This is used to specify the name of the ServletExec/AS instance and is used only for display in the Admin UI to identify the instance.
`-home <path to the ServletExec/AS instance home directory>`	**Optional**. The default is the directory from which the `java` command is executed. This specifies the directory in which the ServletExec/AS instance will look for the `ServletExecData`, `Servlet Logs`, and `Servlets` subdirectories.
`-log <path to the ServletExec/AS instance log directory>`	**Optional**. This can be used to specify an alternate directory for writing the ServletExec/AS log files. By default, log files are written to the `Servlet Logs` subdirectory of the ServletExec/AS home directory as specified by the `-home` argument.

`-root <path to the web server's document root directory>`	**Required**. It specifies the path to the web server's document root directory. If ServletExec/AS is being used with virtual servers, the document root directory of each virtual server must be configured using a separate `-root` argument using the following format:
	
	    `-root <virtual server>=<document root directory>`
	
	for example:
	
	`-root www.abc.com=C:\Apache\htdocs\abcdocs`

`-root <path to the virtual server's document root directory>`	*See previous explanation.*
`-addl <url=directory>`	**Optional**. This can be used to specify additional document directories (other than the root) and is specified in two forms: `-addl <url>=<physical directory>`

Argument	Explanation

	`-addl <virtual server/url>=<physical directory>`
	
	 For example:
	
	 `-addl /test=C:\Test`
	 `-addl www.abc.com/support=C:\Support`

`-addl <virtual server/url=directory>`	*See previous explanation.*
`-mimetypes <path to the web server's mime.types file>`	**Optional**. This specifies the path to the NES, iWS, or Apache `mime.types` file. If this argument isn't provided, the `ServletContext.getMimeType()` method always returns `null`. (For Microsoft IIS, which doesn't have a `mime.types` file, the `ServletContext.getMimeType()` method always returns `null`).
`-allow <ip1,ip2,…,ipn>`	**Optional**. This specifies the address(es) of the web server(s) that are allowed to communicate with the ServletExec/AS instance. An IP address can include the "`*`" character to indicate a subrange (for example: `168.121.97.*`). If the `-allow` argument is not specified, the ServletExec/AS instance only accepts requests from web servers running on the local machine.

Table 6. Start ServletExec/AS Command Line Arguments

5.6.1.3 Closing the DOS Window

When starting a ServletExec/AS instance manually using the `java` command (either using the `StartServletExec.bat` file or from the command line), the DOS window must remain open while the ServletExec/AS instance is running. To be able to close the DOS window, use the `javaw` command instead of `java`. After the ServletExec/AS instance has started, you can close the DOS window and the instance will continue running.

5.6.2 Stopping ServletExec/AS

You must stop a ServletExec/AS instance using the `StopServletExec.bat` file or from the `Shutdown` page of the ServletExec Admin UI. If a ServletExec/AS instance is not shut down properly, servlet `destroy()` methods are not invoked, sessions are not saved, and buffered log messages are lost.

See the ServletExec User Guide for more information about the ServletExec Admin UI.

5.6.2.1 StopServletExec.bat

The batch file `StopServletExec.bat` is provided for your convenience in stopping the ServletExec/AS instance. It is also used by the **Services** control panel to stop the ServletExec/AS instance. `StopServletExec.bat` invokes the `StopServletExec` Java program to stop the ServletExec/AS instance.

`StopServletExec.bat` includes defaults for the `java` and `StopServletExec` command line arguments. You can edit `StopServletExec.bat` to modify these command line arguments, *which you must do when setting or changing the ServletExec Admin UI username and password.* A complete list of `StopServletExec` command line arguments is provided below.

5.6.2.2 Using the java Command

We recommend that you use the `StopServletExec.bat` command file or **Services** to stop a ServletExec/AS instance. However, it is possible to stop a ServletExec/AS instance by invoking the `StopServletExec` program using the `java` command from a DOS command prompt. The information in this section can also help you edit the `StopServletExec.bat` file to modify the parameters to the `StopServletExec` program, *which you must do when setting or changing the ServletExec Admin UI username and password.*

To invoke the `StopServletExec` program using the `java` command

- From the command line, type the following:
    ```
    java [java arguments] StopServletExec [StopServletExec arguments]
    ```

Stop ServletExec/AS arguments

If an option value contains spaces then the entire value must be placed inside double quotes, as in the following example:

```
-name "Local Instance"
```

See the following table for a complete list of available arguments.

Argument	Explanation
`-help`	Displays all available arguments and explanations.
`-host <hostname:port>` or `<ip addr:port>`	Specifies the address of a web server that has a ServletExec/AS web server adapter installed that is configured to communicate with the ServletExec/AS instance.
`-direct <hostname:port>` or `<ip addr:port>`	Specifies the address of the ServletExec/AS instance. Either the `-host` or `-direct` parameter may be specified, but not both. If neither is specified, then the `StopServletExec` program assumes `-host` using the local IP address and port 80. If `-host` is specified without a port, then the `StopServletExec`

Argument	Explanation
	program assumes port 80. If `-direct` is specified without a port, then the `StopServletExec` program assumes port 8888.
`-admin <username/password>`	Specifies the username/password as configured by the ServletExec Admin UI. If these are not specified correctly, the stop command will be rejected by the ServletExec/AS instance. Therefore, you must edit the `StopServletExec.bat` file to enter or modify the `-admin` parameter whenever you set or modify the ServletExec Admin UI username or password.

Table 7. Stop ServletExec/AS Command Line Arguments

5.6.3 Remote Operation

By default, a ServletExec/AS web server adapter is installed on the same machine as the ServletExec/AS instance, and the ServletExec/AS instance only accepts requests from web server adapters running on the local machine. It is possible to install and configure web server adapters to run on a different machine (a "remote" machine) than the ServletExec/AS instance.

To run a web server adapter on a different machine

1. Edit the `StartServletExec.bat` file for the ServletExec/AS instance to add the following argument at the end of the `java` command line:
   `-allow <ip address>`

   where `<ip address>` is the IP address of the remote machine on which the web server adapter will be installed

   See paragraph 5.6.1.2 for a complete discussion of the –allow argument.

2. Run the ServletExec/AS installer on the remote machine on which the web server adapter is to be installed. Choose **Install a web server adapter** when prompted.

   See Section 5.4 for more information.

3. Enter the **name**, **IP address**, and **port number** of the ServletExec/AS instance when prompted.

Note

You can install multiple web server adapters on multiple machines or on a single machine to forward requests to a single ServletExec/AS instance.

5.6.4 Multiple ServletExec/AS Instances (Multiple Java VMs)

It's possible to install and run multiple ServletExec/AS instances on a single server machine. Each ServletExec/AS instance will run independently of the others, each within its own Java Virtual Machine (VM).

To install multiple ServletExec/AS instances

- Run the ServletExec/AS installer again to install another instance. Remember to give each instance a unique name and port number during the installation process.

It's possible to configure a web server adapter to forward HTTP requests to more than one ServletExec/AS instance.

To configure a web server adapter to forward HTTP requests to more than one instance

- Specify the incoming requests to forward using this criteria:
  1. The virtual host to which the request is addressed (i.e., "www.abc.com" or "www.xyz.com")
  2. The servlet alias (if any) specified by the request
  3. The web application context URL (if any) specified by the request

To configure a web server adapter to forward HTTP requests to more than one instance based on a virtual host or servlet alias

- Manually edit the web server configuration file.

  See either the Virtual Servers chapter or the Servlet Alias section in the ServletExec User Guide for detailed instructions.

You can use the ServletExec/AS installer to automatically update a web server configuration file to force requests that specify a web application URL context path to be forwarded to a particular ServletExec/AS instance.

To have requests that specify a web application URL context path forwarded to a particular instance

1. Run the ServletExec/AS installer and select **Install or Update a web server adapter**.
2. Enter the name of a previously installed ServletExec/AS instance when prompted.
3. Specify the web application URL context path when prompted.

   See the ServletExec User Guide for additional information about configuring web applications.

5.7 Uninstalling ServletExec/AS

To uninstall ServletExec/AS or a web server adapter

1. Open Control Panel and click Add/Remove Programs.
2. In the list, click **ServletExec 4.1 AS**, and then click **Add/Remove** or **Change/Remove**, depending on your operating system. This launches the ServletExec installer/uninstaller.

3. Click the appropriate option from the following choices: uninstall a ServletExec/AS instance, uninstall a web server adapter, or completely uninstall ServletExec/AS.

4. If you are uninstalling a web server adapter (or completely uninstalling ServletExec/AS), you must manually edit the web server configuration file(s) to remove the ServletExec/AS directives

    If you are completely removing ServletExec/AS from your system, delete the `ServletExec AS` directory.

    See Section 5.5 for more information.

6. ServletExec/AS: Application Server (UNIX/Linux)

Installing the UNIX web server adapters for Netscape Enterprise Server, iPlanet Web Server, and Apache HTTP Server

T he ServletExec Application Server (ServletExec/AS) is a standalone web application server that hosts Java Servlets and JavaServer Pages (JSP). ServletExec/AS does not contain a built-in HTTP (web) server. Therefore, it must be used in conjunction with Netscape Enterprise Server (NES), iPlanet Web Server (iWS), or Apache HTTP Server. Web server adapters for these web servers are included in the ServletExec/AS installer.

The ServletExec/AS for UNIX® installer is named `ServletExec_AS_41.sh`. If you haven't already done so, you can download the installer from:

> **http://www.newatlanta.com/downloads/index.jsp**

This chapter contains important information that will allow you to verify your installation of ServletExec/AS for UNIX. It will also be useful should you decide to uninstall ServletExec/AS and need to make sure you've completely removed all installed components.

6.1 Upgrading From a Previous Version

You can upgrade to ServletExec 4.1 from all previous versions except versions 3.0C or 3.0E. If you have either of these versions installed, you will need to completely uninstall that version and do a clean installation of ServletExec 4.1.

ServletExec/AS 3.0 and later replaces the ServletExec/Apache and ServletExec/WAI implementations of ServletExec 2.2 and earlier. However, the ServletExec/AS configuration files are slightly different from those for ServletExec/Apache and ServletExec/WAI. Therefore, it is not possible to automatically upgrade from ServletExec/Apache or ServletExec/WAI to ServletExec/AS. Instead, you must do a clean install of ServletExec/AS and manually re-enter your configuration data using the ServletExec Admin UI.

To upgrade to ServletExec/AS 4.1 from 3.0 or 3.1 and maintain your old configuration settings

1. Stop your web server. If you've installed ServletExec/AS adapters for multiple web servers, stop them all.
2. Make backup copies of the `ServletExec Data`, `Servlets`, `classes` (if you've added an additional one), and `webapps` (versions 3.1 and above only) directories for all ServletExec/AS instances.

   ▪ For ServletExec/AS 3.0 and later, the default locations are within:

      `/usr/local/NewAtlanta/ServletExecAS/se-<instance-name>`

3. Close all open applications.
4. Uninstall the old version of ServletExec/AS by running the installer for that version. Choose **remove ServletExec/AS completely** when prompted by the installer.
5. Run the ServletExec/AS 4.1 installer. You will need to re-run the installer for each ServletExec/AS instance, installing into the same location as the previous installation..

   ### Note

   The default installation directory for versions 2.2 and earlier is different.

   ### Important

   For a successful upgrade, you must install everything into the same directory as the previous version.

6. Copy the `ServletExec Data`, `Servlets`, `classes` (if you've added an additional one), and `webapps` (versions 3.1 and above only) directories to the new `ServletExecAS` directory created by the installer in Step 5. In ServletExec/AS 4.1, the default location of the `ServletExecAS` directory is:

      `/usr/local/NewAtlanta/ServletExecAS`

7. Restart your web server.

After restarting your web server, the new version of ServletExec/AS will run using your old configuration settings.

Tip

If you have any problems, you can restore the subdirectories from the backups you made in Step 3.

6.2 System Requirements

6.2.1 Operating System and Web Server

ServletExec/AS 4.1 for UNIX supports the following operating systems and web servers. Again, it supports these web servers using web adapters.

OS	Web Server
SPARC Solaris 2.6, 7, and 8	▪ Netscape Enterprise Server (NES) 3.5.1 and higher ▪ iPlanet Web Server (iWS) 4.0 and higher ▪ Apache HTTP Server 1.3.4 or higher
HP-UX 10.20 and 11.0	▪ Same as SPARC Solaris
AIX 4.3.3 and 4.3.10	▪ IPlanet Web Server (iWS) 4.0 and higher ▪ Apache HTTP Server 1.3.4 or higher
Red Hat Linux 6.2 and 7.0	▪ Apache HTTP Server 1.3.4 or higher

Table 8. AS-UNIX Supported Operating Systems and Web Servers

Notes

▪   In addition to the tested and supported UNIX operating system versions, ServletExec/AS should work on any UNIX variant when used with Apache 1.3.4 or higher (not NES nor iWS). This is accomplished by providing the ServletExec web server adapters for the Apache HTTP Server in source-code form and utilizing the DSO capability of Apache. Because there are a number of potential manufacturer variants and/or release differences in UNIX systems that may cause the installation to fail, ServletExec is not fully supported in these environments. Correcting these types of problems come with issues that are both technical and/or business-related and therefore are not officially supported. If you do have any problems, first contact ServletExec Tech Support at **support@newatlanta.com**. Be sure to include the installation log with your email so that the problems may be accurately analyzed.

▪   In addition to a JDK 1.2-, or 1.3-compliant VM, these installations require an ANSI C compiler (GNU version 2.8.1 or higher preferred. Vendor-specific C compilers are not supported) and a system-specific dynamic shared object loader.

See Section 6.4 for more information on the installation log.

6.2.2 JDK or JRE

ServletExec/AS 4.1 is not supported on JDK or JRE 1.1.x or earlier.

To install ServletExec/AS 4.1, you must first install a JDK 1.2- or 1.3-compliant VM. Visit your operating system vendor's web site for information about downloading and installing a Java VM.

6.2.3 Web server adapters

ServletExec/AS includes web server adapters for the following web servers on UNIX:

- Netscape Enterprise Server 3.5.1 and higher
- iPlanet Web Server 4.0 and higher

- Apache HTTP Server 1.3.4 and higher
- Covalent Raven 1.4 and higher
- C2Net (Red Hat) Stronghold 2.3.x and 2.4.x

6.2.3.1 Apache HTTP Server

Building the ServletExec/AS web server adapter for Apache requires use of the Apache **apxs** utility, which in turn requires that you have **perl**, a **C** compiler, and a dynamic shared linker installed prior to running the installer.

Apache HTTP Server must be built with Dynamic Shared Object (DSO) support enabled before installing a ServletExec/AS web server adapter.

Important

If you have previously built Apache without DSO enabled, you'll also need to rebuild Apache's **apxs** utility. Perform the following steps to rebuild Apache and **apxs** with DSO enabled:

```
# cd <apache source directory>

# rm src/support/apxs

# ./configure --prefix=/usr/local/apache --enable-module=so

# make

# make install
```

6.3 Uninstalling Other Servlet Engines

Very Important

You **must** uninstall any other servlet engines previously installed before installing and using ServletExec/AS. In particular, modifications to the `obj.conf` and `httpd.conf` configuration files for other servlet engines must be removed.

- For Netscape Enterprise Server and iPlanet Web Server, make sure the internal servlet and JSP support is disabled.
 See Section 3.3 for information on disabling servlet engines.

- For Apache HTTP Server, any directives associated with other servlet engines in `httpd.conf` must be removed.
 See Section 5.5.5.2 for more information on `httpd.conf` entries.

6.4 Running the ServletExec/AS Installer

The file `ServletExec_AS_41.sh` is a non-graphical installation shell script.

To use the ServletExec/AS Installer

1. Give the file `execute` permission.

   ```
   # chmod +x ServletExec_AS_41.sh
   ```

2. From a shell command line, run it as follows:

```
# ./ServletExec_AS_41.sh
```

3. Review the ServletExec License Agreement carefully and check the README displayed by the installer for any last minute notices before continuing.
4. Continue following the prompts and instructions until you see confirmation of a successful installation.

You will be able to see a record of the completed installation steps and errors in the log file `/tmp/SEinstall.log`. Should you have any problems with your installation, please include this log file in a message to the ServletExec Technical Support at support@newatlanta.com.

Note

The log file is overwritten each time the installer runs.

6.5 What Was Installed and/or Modified?

When you installed ServletExec/AS, these changes were made to your system:

- The `ServletExecAS` directory was created.
- The web server configuration file was updated to support the ServletExec/AS web server adapter.
- The web server configuration was modified to enable Connection Pooling between the web server adapter and ServletExec.

The following sections describe each of these changes.

6.5.1 The ServletExecAS Directory

The `ServletExecAS` directory was created within the `/usr/local/NewAtlanta` directory, or the directory you specified during the installation.

Important

Do not move the `ServletExecAS` directory after installation. If you move this directory, ServletExec/AS will be unable to find its configuration files.

There are no restrictions on the location of the `ServletExecAS` directory.

The `ServletExecAS` directory contains the following subdirectories:

`bin`	The `bin` subdirectory contains the native code web server adapters used by ServletExec/AS. It also contains utility programs for installing and uninstalling a ServletExec/AS instance as a Windows service.
	Important
	Do not modify the contents of this directory!
`Documentation`	This subdirectory contains the ServletExec User Guide, and the Servlet 2.3 and JSP 1.2 API documentation from

Sun Microsystems.

Examples	This subdirectory contains examples of a web application and a JSP page with an embedded bean.
	See the appropriate chapters of the ServletExec User Guide for instructions on running these examples.
se-*<instance-name>*	There is one se-*<instance-name>* subdirectory for each ServletExec/AS installed on the computer.
	See further discussion of this directory directly below.
installerFiles	The installerFiles subdirectory contains files used by the ServletExec/AS setup program for installing multiple ServletExec/AS instances. This directory and its contents are only used by the installer.
lib	The lib subdirectory contains the Java archive (.jar) files required by ServletExec/AS.

Important

Do not modify the contents of this directory!

The se-*<instance-name>* subdirectories of the ServletExecAS directory each contain the following subdirectories.

classes	The classes subdirectory is automatically added to the ServletExec VM classpath. Place class files that will be shared by multiple servlets in this directory.
	See the README file within the classes subdirectory for more information.
ServletLogs	This subdirectory contains the Servlet.log files that capture the output from the servlet log() method. In some cases, this directory doesn't get created until the first ServletExec initialization.
ServletExecData	This subdirectory contains the ServletExec/AS configuration files. These files are generally modified using the ServletExec Admin UI, but can also be edited manually.
Servlets	This subdirectory contains the class files for servlets hosted by ServletExec/AS.
	See the README in the Servlets directory and the ServletExec User Guide for more information.

webapps   One of these subdirectories exists for each virtual server
                 configured. This subdirectory contains the "default"
                 folder for automatically deploying web applications
                 hosted by ServletExec/AS.

                 *See the README in the webapps directory and the
                 ServletExec User Guide for more information.*

6.5.2 Netscape Enterprise Server / iPlanet Web Server

If you installed a web server adapter for Netscape Enterprise Server (NES) or iPlanet
Web Server (iWS), changes were made to your system as described in the following
sections.

6.5.2.1 Server Configuration File (`obj.conf`)

To install a ServletExec/AS web server adapter for NES/iWS, modifications must be
made to the web server's `obj.conf` file. This section describes the modifications to
`obj.conf` made by the ServletExec/AS installer.

Important

If you chose not to allow the installer to make these modifications, you must make them
manually.

Several lines must be added to `obj.conf` for each server for which ServletExec/AS is
installed (the location of these lines within the `obj.conf` file is very important):

1. Add the following lines to the beginning of `obj.conf` before the other `Init`
   directives:
   ```
   Init            fn="load-modules"
                   shlib="<path>/bin/nsapi/ServletExec_Adapter.dll"
                   funcs="ServletExecInit,ServletExecService"
   Init fn="ServletExecInit" <instance-name>.instances="127.0.0.1:8888"
   ```

   where <path> is the full path to the ServletExec/AS installation directory; by default,
   this is `/usr/local/NewAtlanta/ServletExecAS`.

Note:

The first `Init` directive will normally appear on a single line within the
`obj.conf` file. It's shown as spanning three lines here for formatting reasons. It
may span three lines within `obj.conf`, in which case the second and third lines
must begin with a tab or space character). In general, the format of the second
`Init` directive is:

```
Init fn=ServletExecInit <instance-name>.instances="<addr>:<port>"
```

where *<instance-name>*, <addr>, and <port> are the name, IP address, and port
number of the ServletExec/AS instance to which HTTP requests will be forwarded.

Important—iWS 6.0 or Higher Users

If you're using iWS 6.0 or higher, you must add the `Init` directives to the
`magnus.conf` file rather than the `obj.conf` file.

---

2. The following directives must be added after the <Object name=default> directive:

```
<Object name="<instance-name>">
Service fn="ServletExecService" group="<instance-name>"
</Object>
```

In general, an `Object` directive must be added to `obj.conf` for every ServletExec/AS instance to which the web server adapter will forward requests. The `Object` directives have the form...

```
<Object name="<instance-name>">
Service fn="ServletExecService" group="<instance-name>"
</Object>
```

where `<instance-name>` is the name of the ServletExec/AS instance.

3. Lines similar to the following must be added within the `<Object name=default>` directives:

```
NameTrans fn="assign-name" from="/servlet/*" name="<instance-name>"
NameTrans fn="assign-name" from="*.jsp*" name="<instance-name>"
```

These `NameTrans` directives define URL prefix and suffix mappings that cause HTTP requests to be forwarded to a ServletExec/AS instance for processing.

Figure 17 shows a complete `obj.conf` file for NES 3.6 with the ServletExec/AS directives highlighted.

Important

You must stop and restart the NES/iWS web server after modifying the `obj.conf` configuration file.

```
Init fn="load-modules"
    shlib="/usr/local/NewAtlanta/ServletExecAS/bin/ServletExecAdapter.so"
    funcs="ServletExecInit,ServletExecService"

Init fn="ServletExecInit" <instance-name>.instances="127.0.0.1:8888"

Init fn=flex-init
    access="/usr/netscape/suitespot/https-<instance-name>/logs/access"
    format.access="%Ses->client.ip% - %Req->vars.auth-user% [%SYSDATE%] \"%Req->reqpb.clf-request%\" %Req-
>srvhdrs.clf-status% %Req->srvhdrs.content-length%"

Init fn=load-types mime-types=mime.types

<Object name=default>
NameTrans fn="assign-name" from="/example/*" name="<instance-name>"
NameTrans fn="assign-name" from="/servlet/*" name="<instance-name>"
NameTrans fn="assign-name" from="*.jsp*" name="<instance-name>"
NameTrans fn=pfx2dir from=/ns-icons dir="/usr/netscape/suitespot/ns-icons"
NameTrans fn=pfx2dir from=/mc-icons dir="/usr/netscape/suitespot/ns-icons"
NameTrans fn="pfx2dir" from="/help" dir="/usr/netscape/suitespot/manual/https/ug"
NameTrans fn=document-root root="/usr/netscape/suitespot/docs"
PathCheck fn=nt-uri-clean
PathCheck fn="check-acl" acl="default"
PathCheck fn=find-pathinfo
PathCheck fn=find-index index-names="index.html,home.html"
ObjectType fn=type-by-extension
ObjectType fn=force-type type=text/plain
Service method=(GET|HEAD) type=magnus-internal/imagemap fn=imagemap
Service method=(GET|HEAD) type=magnus-internal/directory fn=index-common
Service method=(GET|HEAD) type=*~magnus-internal/* fn=send-file
AddLog fn=flex-log name="access"
</Object>

<Object name=cgi>
ObjectType fn=force-type type=magnus-internal/cgi
Service fn=send-cgi
</Object>

<Object name="<instance-name>">
Service fn="ServletExecService" group="<instance-name>"
</Object>
```

Figure 17. Complete `obj.conf` Configuration file for NES 3.6

Connection Pooling Between the Web Server Adapter and ServletExec

The installer automatically enables connection pooling between the web server adapter and ServletExec, using default settings. The purpose of this connection pooling is to improve performance through better resource management.

You can change the default settings by modifying the configuration file as explained below. The two parameters that provide the connection pooling are: `pool max idle` (sets maximum idle time for a connection… default setting is 10) and `pool increment`. (sets connection increment value… default setting is 5).

A good guideline for changing these parameters from their default settings would be to set the `pool max idle` to your average number of connections, and to set the `pool increment` to the average spike number of connections.

To modify Connection Pooling in NES/iWS for ServletExec/AS

- Add the following to the beginning of the `obj.conf` file. Go back to Step 1 earlier in this section, copy the line containing the second init directive and insert a space at the end of it, adding the following immediately after and on the same line:

```
<instance-name>.pool-max-idle=<value> <instance-name>.pool-
increment=<value>
```

6.5.2.2 Web Application URL Context

For each web application URL context configured using the ServletExec Admin UI, you must add a *NameTrans* directive to the `obj.conf` file.

To update the applications property in `obj.conf`

Use the following form:

```
NameTrans fn="assign-name" from="/URL-context/*" name="<instance-name>"
```

6.5.2.3 Prefix Aliases

For each prefix alias configured using the ServletExec Admin UI, you must add a *NameTrans* directive to in the `obj.conf` file.

To update the aliases property in `obj.conf`

Use the following form:

```
NameTrans fn="assign-name" from="/prefix-alias/*" name="<instance-name>"
```

6.5.2.4 Suffix Aliases

For each suffix alias configured using the ServletExec Admin UI, you must add a *NameTrans* directive to the `obj.conf` file.

To update the aliases property in `obj.conf`

Use the following form:

```
NameTrans fn="assign-name" from="/suffix-alias/*" name="<instance-name>"
```

6.5.3 Apache HTTP Server

If you installed a web server adapter for Apache, changes were made to your system as described in the following sections.

6.5.3.1 mod_servletexec.so

The `mod_servletexec.so` library implements the ServletExec/AS web server adapter for Apache HTTP Server. It does so after the ServletExec/AS installer copies it to the `libexec` subdirectory of the Apache installation directory.

6.5.3.2 Server Configuration File (httpd.conf)

To install a ServletExec/AS web server adapter for Apache, modifications must be made to the web server's `httpd.conf` file. This section describes the modifications to `httpd.conf` made by the ServletExec/AS installer.

Important

If you chose not to allow the installer to make these modifications, you must make them manually.

The following directive causes Apache to load the ServletExec/AS web server adapter and was added to `httpd.conf` with the other `LoadModule` directives (near the top of the file):

```
LoadModule servletexec_module libexec/mod_servletexec.so
```

The following directives define the default prefix and suffix aliases used by ServletExec/AS and were added to the end of the `httpd.conf` file:

```
ServletExecInstances <instance-name> 127.0.0.1:8888
ServletExecAliases <instance-name> /servlet .tmpl .jsp

<Location /servlet>
    SetHandler servlet-exec
</Location>

AddHandler servlet-exec jsp
```

See the ServletExec User Guide for a general discussion of servlet aliases, as well as Section 6.5.3.3 and Section 6.5.3.5 for further discussion of configuring prefix and suffix aliases with Apache.

Connection Pooling Between the Web Server Adapter and ServletExec

This feature is not currently available for ServletExec/AS on Apache Http Server (UNIX).

6.5.3.3 Web Application URL Context

For each web application URL context configured using the ServletExec Admin UI, you must update the *ServletExecApplications* directive in the `httpd.conf` file.

To update the ServletExecApplications directive in `httpd.conf`

- Use the following form:
  ```
  ServletExecApplications <instance-name> /URL-context1 /URL-context2
  <Location /URL-context1>
                  SetHandler servlet-exec
  </Location>
  <Location /URL-context2>
                  SetHandler servlet-exec
  </Location>
  ```

6.5.3.4 Prefix Aliases

For each prefix alias that is configured using the ServletExec Admin UI, the `ServletExecAliases` directive must be updated, and a `Location` directive must be added to `httpd.conf` in the following format:

```
ServletExecAliases <instance-name> /prefix-alias
<Location /prefix-alias>
    SetHandler servlet-exec
</Location>
```

If you allowed the ServletExec/AS installer to automatically update `httpd.conf`, ServletExec/AS is configured to handle the `/servlet` prefix alias by default.

6.5.3.5 Suffix Aliases

There are two methods for configuring suffix aliases with Apache. The most common method is illustrated above. For each suffix alias that is configured using the ServletExec Admin UI, add an `AddHandler` directive to `httpd.conf` with the following format:

```
ServletExecAliases <instance-name> <suffix-aliases>
AddHandler servlet-exec <suffix-alias>
```

You should use this method if the URL Rewriting option of the Session Tracking feature is disabled, which it is by default. This method of configuring suffix aliases provides better performance than the alternative method.

See the ServletExec User Guide for more information on Session Tracking and URL Rewriting.

URL Rewriting Alternative

If you enabled the URL Rewriting option of the Session Tracking feature, you must use an alternative method of configuring suffix aliases. For this method, you use a `Location` directive instead of the `AddHandler` directive. For each suffix alias that is configured using the ServletExec Admin UI, add a `Location` directive to `httpd.conf` with the following format:

```
<Location /*.suffix-alias*>
SetHandler servlet-exec
</Location>
```

In addition, for each subdirectory level beneath the web server document root directory, you must add an additional `Location` directive to `httpd.conf`. For example, assume the web server root document directory is `/usr/local/apache/htdocs` and the following subdirectories appear beneath `htdocs`:

```
htdocs/subdir1
htdocs/subdir2
htdocs/subdir2/subdir3
```

In this example, there are two levels of subdirectories beneath `htdocs`, so to create a suffix alias of `.jsp`, you should add three `Location` directives to `httpd.conf`, one for `htdocs` and one for each subdirectory level:

```
#
# suffix alias for htdocs
#
<Location /*.jsp*>
SetHandler servlet-exec
</Location>

#
# suffix alias for htdocs/subdir1 and htdocs/subdir2
#
<Location /*/*.jsp*>
SetHandler servlet-exec
</Location>
```

```
#
# suffix alias for htdocs/subdir2/subdir3
#
<Location /*/*/*.jsp*>
SetHandler servlet-exec
</Location>
```

6.6 ServletExec/AS Operation

This section provides instructions for starting and stopping ServletExec/AS, for configuring a ServletExec/AS instance to run on a remote machine from the web server, and for configuring multiple ServletExec/AS instances for a web server.

6.6.1 Starting ServletExec/AS

A ServletExec/AS instance is a standalone Java application that communicates with the web server (through the ServletExec/AS web server adapter) using network sockets. You start a ServletExec/AS instance using the `java` command on the command line. In addition to `java` command arguments, ServletExec/AS accepts several required and optional arguments. You can also start a ServletExec/AS instance using the `StartServletExec` script. The following sections describe both methods.

6.6.1.1 StartServletExec

The `StartServletExec` script is provided for your convenience in starting the ServletExec/AS instance. `StartServletExec` includes defaults for all of the `java` and ServletExec/AS command line arguments. You can edit `StartServletExec` to modify these command line arguments. The next section features a complete list of ServletExec/AS command line arguments.

6.6.1.2 Using the java Command

We recommend that you use the `StartServletExec` script to start a ServletExec/AS instance. However, it is possible to start a ServletExec/AS instance from the command line using the `java` command. The information in this section can also help you edit the `StartServletExec` script to modify the startup parameters for a ServletExec/AS instance.

To start a ServletExec/AS instance using the `java` command

- From the command line, type the following:

      java [java arguments] ServletExec [ServletExec/AS arguments]

  Important

  The `java` command arguments must include the `–classpath` option specifying the path to the `ServletExec41.jar`, `servlet.jar`, `crimson.jar`, `jaxp.jar`, and `jndi.jar` archives.

Start ServletExec/AS arguments

If an option value contains spaces then the entire value must be placed inside double quotes, as in the following example:

```
-name "Local Instance"
```

The following table lists and explains all ServletExec/AS arguments.

Argument	Explanation
`-help`	Displays all available arguments and explanations.
`-port <port number>`	**Optional**. The default value is 8888. This argument specifies the TCP/IP port on which the ServletExec/AS instance communicates with the web server adapter(s). If you modify this value, you must also modify the web server configuration file(s) as described above.
`-backlog <length>`	**Optional**. The default value is 50. This argument specifies the size of the ServletExec/AS incoming request queue. Requests received when the queue is full are rejected. Increasing the size of the request queue may prevent requests from being rejected under heavy loads, but may increase per-request response times.
`-name <instance-name>`	**Optional**. This is used to specify the name of the ServletExec/AS instance and is used only for display in the Admin UI to identify the instance.
`-home <path to the ServletExec/AS instance home directory>`	**Optional**. The default is the directory from which the `java` command is executed. This specifies the directory in which the ServletExec/AS instance will look for the `ServletExecData`, `ServletLogs`, and `Servlets` subdirectories.
`-log <path to the ServletExec/AS instance log directory>`	**Optional**. This can be used to specify an alternate directory for writing the ServletExec/AS log files. By default, log files are written to the `ServletLogs` subdirectory of the ServletExec/AS home directory as specified by the `-home` argument.
`-root <path to the web server's document root directory>`	**Required**. It specifies the path to the web server's document root directory. If ServletExec/AS is being used with virtual servers, the document root directory of

Argument	Explanation
	each virtual server must be configured using a separate -root argument using the following format: -root <virtual server>=<document root directory> for example: -root www.abc.com=/usr/local/Apache/htdocs/abcdocs
-root <path to the virtual server's document root directory>	*See previous explanation.*
-addl <url=directory>	**Optional**. This can be used to specify additional document directories (other than the root) and is specified in two forms: -addl <url>=<physical directory> -addl <virtual server/url>=<physical directory> For example: -addl /test=usr/local/test -addl www.abc.com/support=usr/local/support
-addl <virtual server/url=directory>	*See previous explanation.*
-mimetypes <path to the web server's mime.types file>	**Optional**. This specifies the path to the NES, iWS, or Apache mime.types file. If this argument isn't provided, the ServletContext.getMimeType() method always returns null.
-allow <ip1,ip2,…,ipn>	**Optional**. This specifies the address(es) of the web server(s) that are allowed to communicate with the ServletExec/AS instance. An IP address can include the "*" character to indicate a subrange (for example: 168.121.97.*). If the -allow argument is not specified, the ServletExec/AS instance only accepts requests from web servers running on the local machine.

Table 9. Start ServletExec/AS Command Line Arguments

6.6.2 Stopping ServletExec/AS

You must stop a ServletExec/AS instance using the StopServletExec script or from the Shutdown page of the ServletExec Admin UI. If a ServletExec/AS instance is not

shut down properly, servlet `destroy()` methods are not invoked, sessions are not saved, and buffered log messages are lost.

See the ServletExec User Guide for more information on the ServletExec Admin UI.

6.6.2.1 StopServletExec

The `StopServletExec` script is provided for your convenience in stopping the ServletExec/AS instance. `StopServletExec` invokes the `StopServletExec` Java program to stop the ServletExec/AS instance.

`StopServletExec` includes defaults for the `java` and `StopServletExec` command line arguments; you can edit `StopServletExec` to modify these command line arguments, *which you must do when setting or changing the ServletExec Admin UI username and password.* A complete list of `StopServletExec` command line arguments is provided below.

6.6.2.2 Using the java Command

We recommend that you use the `StopServletExec` script to stop a ServletExec/AS instance. However, it is possible to stop a ServletExec/AS instance by invoking the `StopServletExec` program using the `java` command from a command line. The information in this section can also help you edit the `StopServletExec` script to modify the parameters to the `StopServletExec` program, *which you must do when setting or changing the ServletExec Admin UI username and password.*

To invoke the `StopServletExec` program using the `java` command

- From the command line, type the following:

      java [java arguments] StopServletExec [StopServletExec arguments]

Stop ServletExec/AS arguments

If an option value contains spaces then the entire value must be placed inside double quotes, as in the following example:

    -name "Local Instance"

The following table lists and explains the available Stop ServletExec/AS arguments.

Argument	Explanation
`-help`	Displays all available arguments and explanations.
`-host <hostname:port>` or `<ip addr:port>`	Specifies the address of a web server that has a ServletExec/AS web server adapter installed that is configured to communicate with the ServletExec/AS instance.
`-direct <hostname:port>` or `<ip addr:port>`	Specifies the address of the ServletExec/AS instance. Either the `-host` or `-direct` parameter may be specified, but not both. If neither is specified, then the `StopServletExec`

Argument	Explanation
	program assumes `-host` using the local IP address and port 80. If `-host` is specified without a port, then the `StopServletExec` program assumes port 80. If `-direct` is specified without a port, then the `StopServletExec` program assumes port 8888.
`-admin <username/password>`	Specifies the username/password as configured by the ServletExec Admin UI. If these are not specified correctly, the stop command will be rejected by the ServletExec/AS instance. Therefore, you must edit the `StopServletExec` script to enter or modify the `-admin` parameter whenever you set or modify the ServletExec Admin UI username or password.

Table 10. Stop ServletExec/AS Command Line Arguments

6.6.3 Remote Operation

By default, a ServletExec/AS web server adapter is installed on the same machine as the ServletExec/AS instance, and the ServletExec/AS instance only accepts requests from web server adapters running on the local machine. It is possible to install and configure web server adapters to run on a different machine (a "remote" machine) than the ServletExec/AS instance.

To run a web server adapter on a different machine

1. Edit the `StartServletExec` script for the ServletExec/AS instance to add the following argument at the end of the `java` command line:
    `-allow <ip address>`

    where `<ip address>` is the IP address of the remote machine on which the web server adapter will be installed

    See Section 6.6.1.2 for a detailed discussion of the `-allow` argument.

2. Run the ServletExec/AS installer on the remote machine on which the web server adapter is to be installed. Choose **Install a web server adapter** when prompted.

    See Section 6.4 for more information.

3. Enter the **name**, **IP address**, and **port number** of the ServletExec/AS instance when prompted.

Note

You can install multiple web server adapters on multiple machines or on a single machine to forward requests to a single ServletExec/AS instance.

6.6.4 Multiple ServletExec/AS Instances (Multiple Java VMs)

It is possible to configure a web server adapter to forward HTTP requests to more than one ServletExec/AS instance.

To install multiple ServletExec/AS instances

- Run the ServletExec/AS installer again to install another instance. Remember to give each instance a unique name and port number during the installation process.

To configure a web server adapter to forward HTTP requests to more than one instance

- Specify the incoming requests to forward using this criteria:

  1. The virtual host to which the request is addressed (i.e., "www.abc.com" or "www.xyz.com")

  2. The servlet alias (if any) specified by the request

  3. The web application context URL (if any) specified by the request

To configure a web server adapter to forward HTTP requests to more than one instance based on a virtual host or servlet alias

- Manually edit the web server configuration file.

  See the Virtual Servers chapter and/or the Servlet Alias section in the ServletExec User Guide for detailed instructions.

  ### Note
  You can use the ServletExec/AS installer to automatically update a web server configuration file to cause requests that specify a web application URL context path to be forwarded to a particular ServletExec/AS instance.

To have requests that specify a web application URL context path forwarded to a particular instance

  1. Run the ServletExec/AS installer and select **Install or Update a web server adapter**.

  2. Enter the name of a previously installed ServletExec/AS instance when prompted.

  3. Specify the web application URL context path when prompted.

  See Chapter 3 in the ServletExec User Guide for additional information on configuring web applications.

6.7 Uninstalling ServletExec/AS

To uninstall ServletExec/AS or a web server adapter

1. Re-run the ServletExec/AS installation script.
2. Click the appropriate option from the following choices: uninstall a ServletExec/AS instance, uninstall a web server adapter, or completely uninstall ServletExec/AS.
3. If you are uninstalling a web server adapter or completely uninstalling ServletExec/AS, you must manually edit the web server configuration file (`obj.conf` for NES/iWS, or `httpd.conf` for Apache) to remove any changes you may have made for ServletExec/AS.